Evaluación y comparación de tecnologías de movilidad IP sobre servidores

Proyecto de Sistemas Informáticos 2010/2011

Autores:
Markel Arizaga Bégil
Ibai Irastorza Azkarate
Javier Nobel Antón

Director:
Rafael Moreno Vozmediano

16/09/2011

AUTORIZACIONES

Autorizamos a la Universidad Complutense de Madrid a utilizar y/o difundir con fines académicos y no comerciales, siempre mencionando expresamente a sus autores, tanto la propia memoria como el código, la documentación y/o el prototipo desarrollado.

Markel Arizaga Bégil

Ibai Irastorza Azkarate

Javier Nobel Antón

AGRADECIMIENTOS

Queremos agradecer a nuestro director de proyecto, el profesor Rafael Moreno Vozmediano por la ayuda que nos ha prestado para poder realizar este proyecto, y por estar siempre disponible cada vez que le hemos necesitado. Sin él, no habríamos podido sacar a delante todo este trabajo.

También agradecer a nuestros familiares y amigos la comprensión por tantas horas invertidas en este proyecto y no dedicadas a ellos. Sin su apoyo nada tendría sentido.

ABSTRACT

When Internet was created, it was targeted to connect remote machines between them, even if those where separated by thousands of kilometers. To do that, different technologies and protocols where developed and still nowadays these technologies and protocols are in force. However, when mobile devices arrived the needs of the users of the network have changed and these new needs require the old technologies and protocols to be upgraded to match with the modern times in order to give coverage to one of the sectors with the biggest progression in the next years, mobility.

Therefore, different proposals have been started to be developed in order to give mobility support to the internet and cover the new needs of users.

In the document below, the authors evaluate different proposals to achieve IP mobility framed in their Computer Engineering Carrier Ending Project, taken at the Complutense University of Madrid.

RESUMEN

Cuando Internet fue diseñada, ésta estaba destinada a conectar entre sí máquinas que se encontraban a muchos kilómetros de distancia. Para ello se desarrollaron distintas tecnologías y protocolos que todavía hoy siguen en funcionamiento. Sin embargo, con la llegada de los dispositivos móviles las necesidades de los usuarios de la red han cambiado y con ellas, los protocolos de internet ven necesario un amoldamiento a los tiempos actuales para dar cobertura a uno de los sectores con más progresión para los próximos años, la movilidad.

Visto esto, ya se han comenzado a desarrollar distintas propuestas para que los protocolos de comunicación soporten la movilidad de forma que las nuevas necesidades queden cubiertas.

En el presente documento se evaluarán distintas propuestas para la movilidad IP, en el marco del proyecto de fin de carrera de Ingeniería en Informática, cursada por los autores del presente documento en la Universidad Complutense de Madrid.

ÍNDICE

PALABRAS CLAVE

VPN, HIP, Mobile IP, Movilidad de servidores, redes, conectividad, transferencia de ficheros, LAN, WLAN, TCP/IP

INTRODUCCIÓN

Introducción a las redes

En este apartado, se hará una breve introducción a las redes de computadores o redes informáticas con el fin de introducir al lector en el mundo en el que se ha desarrollado el proyecto que se trata en el presente documento.

Una red de computadores o red informática es un conjunto de equipos informáticos unidos entre sí por un medio físico por el cual se envía información de unos equipos a otros. La finalidad de crear grupos de ordenadores capaces de comunicarse entre sí no es otra que la de compartir recursos e información entre equipos que no necesariamente se encuentren en una misma localización geográfica.

Desde sus inicios, las redes de computadores han ido avanzando en sus implementaciones con el fin de proporcionar unos niveles de seguridad adecuados y una velocidad de transmisión que permita que la comunicación sea viable.

La red de computadores más conocida y utilizada hoy en día es sin duda Internet, una red que a nivel global es capaz de poner en contacto computadores sin importar en qué lugar se encuentren.

Los orígenes de Internet se remontan al año 1969, año en el que se produce la primera conexión entre ordenadores situados en tres universidades de California y una de Utah, todas ellas en los Estados Unidos. Esta primera red es conocida con el nombre de ARPANET.

Sin embargo, en unas pocas décadas, aquella primera red entre universidades se ha convertido en uno de los elementos más importantes para el desarrollo de la educación, economía y las comunicaciones globales.

Obviamente, con el paso de los años, las necesidades que las redes de comunicación debían cubrir han ido cambiando, pasando de ser un mero enlace de documentos de hipertexto a convertirse en la vía de transmisión de todo tipo de contenidos.

Pero no sólo los contenidos que viajan por la red han cambiado, puesto que también lo han hecho las formas de utilizar y consumir dicho contenido. Con el avance tecnológico de los últimos años, la movilidad se ha erigido en uno de los pilares básicos de entrada a Internet, y es a su vez uno de los campos con mayor previsión de crecimiento y desarrollo dentro del sector informático. Es concretamente este campo, el de la movilidad, en el que está enmarcado el proyecto fin de carrera que se especifica en la presente memoria. En el siguiente apartado se hablará más específicamente del campo de la movilidad en internet para pasar después a describir con más exactitud y más exhaustividad los pormenores del proyecto.

Protocolos y arquitecturas de red

Actualmente las máquinas que son capaces de conectarse a una red, se basan en el modelo OSI para funcionar. Éste es un modelo basado en capas en las que cada capa se encarga de una tarea específica a la hora de llevar a cabo una comunicación de red, pasando desde la gestión de más bajo nivel (gestionando elementos físicos de transmisión) hasta el nivel con mayor abstracción.

En realidad, en la actualidad el modelo OSI no es utilizado tal y como está especificado ya que se han ido introduciendo cambios que buscaban mayor flexibilidad en los protocolos de red. Estos cambios han tendido a *diluir* en cierto modo los límites de cada capa para llegar a un modelo que no separa los quehaceres de cada una de forma tan estricta. Aún así, el modelo OSI es muy utilizado en entornos educativos y además será útil más adelante en esta memoria, ya que uno de los protocolos que se estudian en la misma tienen cierto impacto sobre el modelo tradicional.

La siguiente figura muestra las capas de las que se compone el modelo OSI.

Ilustración 1 : Pila del modelo OSI

Como ya se ha dicho anteriormente, todas estas capas interactúan entre sí para poder hacer que un ordenador sea capaz de comunicarse con otros a través de una red. Cada capa solamente se comunica con las capas contiguas, proporcionando o recibiendo la información necesaria para efectuar su tarea y entregar los datos resultantes a la capa siguiente.

Cada una de estas capas tiene unos protocolos propios que las hacen funcionar. Concretamente, las que más van a interesar a lo largo de la memoria serán los protocolos de las capas de red y de transporte. Estas capas, y por ende internet, se sustentan en la familia de protocolos TCP/IP, consiguiendo que las distintas redes heterogéneas sean capaces de funcionar como si de una única red homogénea se tratase. El nombre de TCP/IP lo obtiene de los dos principales protocolos de esta familia, pero en realidad existen decenas de protocolos, cada uno con su objetivo. Algunos de los protocolos más conocidos son Hyper Text Transfer

Protocol (HTTP), File Transfer Protocol (FTP), AddressResolutionProtocol (ARP) o Internet Control MessageProtocol (ICMP).

El correcto funcionamiento de internet se basa en la colaboración de todos estos protocolos, puesto que al combinar el objetivo de cada uno de ellos se consigue que los ordenadores puedan conectarse a la red como lo hacen habitualmente.

Por otra parte, cabe mencionar que existen diferentes arquitecturas de red. Una arquitectura de red especifica que rol cumplen cada uno de los ordenadores de la red. Entre estas arquitecturas, sin lugar a dudas la más extendida es cliente-servidor. En esta arquitectura existe un nodo que se encarga de ofrecer unos recursos concretos, y los clientes serán los que hagan las peticiones al servidor cuando quieran hacer uso de los mencionados recursos.

Un ejemplo de arquitectura que ha ganado popularidad en la última década es la llamada P2P o Peer to Peer en la que no existen servidores ni clientes, sino que todas las máquinas se comportan como iguales. Esta arquitectura ha tenido mucho uso en redes de intercambio de archivos o para usos de comunicaciones como en algunas soluciones de VoIP (Voz sobre IP).

Problemática o estado del arte

Como se ha mencionado en el anterior apartado, uno de los puntos fuertes del sector de la informática es hoy en día el de la conectividad en movilidad. En los últimos años se ha visto un gran avance en los dispositivos móviles, llegando estos a tener en la actualidad la capacidad de computación de los ordenadores de hace algunos años. Este avance ha traído un nuevo tipo de dispositivo capaz de conectarse a la red de redes, y con ello nuevas posibilidades de utilizar los contenidos y servicios de Internet.

Obviamente, este nuevo escenario ha supuesto también la necesidad de adaptación de las mencionadas redes, ya que estas no fueron diseñadas pensando en dispositivos móviles, sino computadores que se mantendrían siempre en un mismo lugar.

El mayor reto al que se ha enfrentado el sector de la movilidad, es el de proporcionar conectividad en cualquier lugar en el que pueda encontrarse el dispositivo móvil, y más importante aún, conseguir mantener las conexiones activas sin importar el cambio de ubicación que pudiera darse durante el tiempo de vida de estas.

En un principio, cuando se ha comenzado a tener en los móviles capacidad real de conectarse a Internet, las conexiones eran posibles, pero no estaba tan bien depurado el factor de cambiar de una red a otra sin que el usuario experimentara interrupciones en el servicio.

Este es el motivo por el cual se han ido desarrollando nuevos protocolos para conseguir que la movilidad en internet sea una realidad factible y con un rendimiento adecuado para poder llevar a cabo todas las actividades que ofrece la red, haciendo lo más transparente posible el hecho de que existe un factor de movilidad.

En el anterior apartado, se ha hablado del protocolo cliente servidor, por el cual existe un computador que actúa como servidor y funciona respondiendo a peticiones que le llegan de los clientes. Este protocolo también es utilizado en el caso del internet móvil, siendo los dispositivos móviles clientes de los servidores habituales a los que puede accederse desde los ordenadores convencionales.

Sin embargo, este proyecto se centra en un punto de vista diferente. Este punto de vista considera que el elemento que está en movilidad es el servidor en vez del cliente. Obviamente, el caso contrario, el de clientes en movilidad ya está siendo desarrollado, puesto que como se ha dicho tiene un futuro muy prometedor. Sin embargo, el tener el servidor en movilidad es un

escenario que no es tan usual como el anterior, y que ha sido el escenario objeto de investigación del proyecto.

El uso de servidores en la actualidad es extendido, por todos los usuarios, ya sea a nivel de empresa para trabajar sobre datos y ficheros con datos comunes, como para consultar información de periódicos o revistas colocadas en servidores, hasta descargar canciones o películas, también almacenadas en servidores. Todo este tipo de interacción entre máquinas e intercambio de información tienen un patrón común: la existencia de un servidor, en el que se almacena la información, y de unos clientes, que hacen uso de ella. Puesto que mucha gente depende de un servidor, es un tema delicado un cambio de red del servidor. Este cambio se puede deber a diversos motivos: sobrecarga de la red en la que se encontraba, balanceo de carga por parte del firewall o router para optimizar la velocidad de la conexión, avería en la red en la que estaba y traslado a otra red de la empresa pero con distinta dirección…. En todos los casos, la dirección IP del servidor tendrá que cambiar. ¿Qué problemas acarrea esta situación?

La dirección IP se encarga de otorgar a la máquina identificación y ubicación. Si se cambia la dirección IP de un servidor para poder colocarlo en otra red, se cambiará su ubicación, que era el objetivo; pero también se estará cambiando la identificación del servidor, por lo que, todos los clientes que tuvieran conexiones realizadas con el servidor no sabrán donde se encuentra este, por lo que la conexión que se interrumpió no podrá ser reanudada. Para que estas conexiones no se pierdan, y solo se detengan momentáneamente durante el cambio de red del servidor, existen diversas técnicas: VPN a nivel software, VPN a nivel hardware, MIP, HIP,….; en todas ellas, el objetivo es, de una forma o de otra, que esa conexión no se pierda. Consiguen que, aunque una máquina se traslade a otra red, la conexión que había con la IP antigua no se pierda, y de una forma u otra, el cliente sea capaz de conocer la nueva ubicación del servidor, y reanudar la transferencia. Sobre esta problemática tratará el proyecto.

Concretamente, el proyecto tratará de evaluar y comparar distintos protocolos y diferentes tecnologías que proporcionan soporte a la movilidad IP sobre servidores, e intentará concluir cuál de los protocolos investigados se comporta mejor, en que situaciones y para qué tipo de servicios.

Los protocolos estudiados son Virtual Private Network (VPN) y Host IdentityProtocol (HIP) de los que se hablará más adelante en esta memoria. También cabe mencionar que en un principio se intentó estudiar un tercer protocolo, llamado Mobile IP, pero que tuvo que ser descartado durante el transcurso del proyecto por causas que se especificarán en la sección referida a este protocolo.

Los siguientes apartados, por tanto, harán una introducción a cada una de los tres protocolos mencionados, dando una visión de cómo funcionan y mostrarán de forma detallada el entorno de trabajo construido para hacer las pruebas y obviamente, información detallada sobre las pruebas realizadas con dichos protocolos.

OBJETIVO

El objetivo del proyecto es la evaluación y comparación de tecnologías de movilidad IP sobre servidores. Dada la problemática con el cambio de IP de los servidores mencionada anteriormente, y el extendido e imprescindible uso de los servidores por parte de todos los usuarios de la red, es necesario poder migrar un servidor de red sin necesidad de perder las conexiones que se tuviera en ese momento, y hacer que el tiempo que tarde en restablecerse la configuración del mismo, así como en levantarse las conexiones sea mínimo.

Por estos motivos surge este proyecto. Dado que hay tecnologías encargadas de esta movilidad, se quiso hacer un estudio detallado de estas, con el objetivo de evaluar y comparar dichas tecnologías.

Como ya se indicó en el estado del arte, existen diversas tecnologías para la movilidad IP aplicables a servidores. En este trabajo se optó por estudiar tres de las técnicas más desarrolladas y avanzadas que existen en este sector, como son VPN, HIP y MIP. Existen multitud de técnicas más, pero se escogieron estas para el desarrollo, pues son de las más extendidas en el mercado, las más comunes, y son diversas entre ellas, y se creyó que era una buena elección para poder tener una buena comparativa.

Para evaluar VPN, se optó por OPENVPN, software gratuito de SSL, que consiste en montar una red virtual encima de la red pública. Con ello, y a través de diversos mecanismos que se explican a continuación, se puede tener un equipo (en este caso un servidor) en una red física, pero que “virtualmente” esté en otra red, teniendo una IP de la misma, y teniendo conexión a través del servidor VPN.

Con la tecnología HIP decidimos utilizar OPENHIP, también software gratuito de SSL. Su estándar más moderno y avanzado es la versión 7.0, y fue sobre la que se desarrolló el proyecto. Hace poco ha salido la versión 8.0, pero todavía está en pruebas. HIP consiste en introducir una nueva capa al protocolo TCP/IP, la capa HIP. Con esta nueva capa, se consigue desligar la identidad y localidad que lleva consigo la dirección IP; así, ahora obtenemos dos direcciones, una con la que se indica la identidad de cada máquina, la cual no cambia, y otra que indica la localidad y ubicación de esta máquina (en este caso, el servidor, que es sobre lo que se desarrolla el proyecto). Con ello, cuando se produce un cambio de red del servidor, la última dirección cambiará, y mediante una serie de actualizaciones, los clientes sabrán el lugar donde se encuentra, por lo que un cambio de ubicación no será algo definitivo para la comunicación, como ocurría con un cambio de la dirección IP.

Por último, en la tecnología MIP,cada nodo móvil tiene dos direcciones IP, home y Care of Address(CoA) que permite la comunicación entre hosts independientemente de la red en la que se encuentren en un momento dado, permitiendo así la movilidad de los nodos. Cuando un pc se mueve, estas direcciones se van actualizando, para tener constancia en cada momento de la ubicación del host, gracias a la separación en estas dos IPs de localización y de identificación. Para estudiar MIP se utilizó Mobile IP Dynamics, de la universidad de Helsinki, dado su carácter estable, jerárquico, y que era gratuito. Después de muchas pruebas, no se logró poner en funcionamiento, y después de hablar con el director del proyecto, se decidió continuar con la investigación sin tener en cuenta esta tecnología.

Una vez claro el problema (evaluación y comparación de tecnologías para movilidad IP de servidores), y las técnicas a utilizar, sólo queda por definir las medidas con las que se realizaría el trabajo. Se eligieron distintas pruebas para poder comparar las tecnologías desde diversos puntos de vista. Estas pruebas fueron:

Ping: Utilidad de diagnóstico por excelencia. Consiste en el envío de paquetes de datos de un host a otro, con el objetivo de ver el tiempo que tarda un host en recibir la petición, y en devolverla. Para esta prueba, se utilizaron diversos tamaños de paquete, y se aumentó la frecuencia de envío, para analizar el comportamiento de las tecnologías.

Peticiones http: el protocolo por excelencia de internet es el http. “Protocolo de transferencia de hipertexto”, el acceso a la información de las páginas web suele desarrollarse por éste método. Con esta prueba se podría comprobar el comportamiento de usuarios lanzando peticiones sobre el servidor para la consulta de hipertexto, y si se recuperaría de la marcha de un servidor de una red a otra mientras se realizan las consultas.

Descarga de un fichero por http: Otro de los usos básicos de internet es la descarga de ficheros a través de las páginas web. Un servidor con ficheros permite la descarga de ficheros a través del protocolo http de diversos clientes. Se quiere estudiar el comportamiento de las conexiones durante el cambio de red de un servidor.

Transferencia de un fichero por scp: Scp es un medio de transferencia segura que usa el protocolo ssh. Esta transferencia se realiza entre dos host en remoto, o entre un host local y otro host remoto. Durante la transferencia de datos, los datos son cifrados para evitar posibles extracciones de la información por agentes externos a la comunicación. Uno de los objetivos de las mediciones era ver que ocurría cuando, en mitad de una descarga, el servidor cambiaba su IP, para observar si se cancelaba la conexión, se paraba un tiempo y luego proseguía, si había que reactivar la conexión, etc...

Transferencia de un fichero por ftp: Ftp es el protocolo de transferencia de ficheros por excelencia. Utiliza para ello habitualmente el puerto 20. Se basa en la transferencia de ficheros entre un servidor y un cliente. El cliente puede coger un fichero desde el servidor, o puede depositar un fichero en el mismo. Para ello, deberá loggearse antes contra el servidor, y una vez que ya esté realizada la conexión, podrá empezar la descarga del fichero. FTP está orientado a conseguir la mayor velocidad de transferencia, pero no es tan seguro como lo es scp. Al igual que antes, se realizará un cambio de la red el servidor durante la descarga, para estudiar su comportamiento.

Escaneo de puertos con hping: Hping es una herramienta para realizar auditorías y poder realizar pruebas sobre una red determinada. Además del envío de paquetes, tanto a nivel tcp, como a nivel udp, tiene una aplicación interesante que todavía no se ha mencionado, y es el escaneo de puertos. Con hping, se puede enviar paquetes a cada puerto, para comprobar qué puertos están abiertos y cuáles no, dependiendo de la respuesta obtenida.

PROTOCOLOS ESTUDIADOS DURANTE EL PROYECTO

Introducción

En este apartado se mencionarán los tres protocolos que han sido objeto de estudio en este proyecto. Para todos ellos, se dará una explicación de cuál es la filosofía de funcionamiento de cada uno de ellos.

Mobile IP

Mobile IP es un protocolo de comunicaciones estándar que permite a nodos móviles saltar de una red a otra manteniendo una dirección IP constante.

Introducción

Este protocolo permite el enrutado de datagramas IP independientemente de la localización geográfica. Para esto, cada nodo móvil tiene dos direcciones IP, home y Care of Address(CoA) que permite la comunicación entre hosts independientemente de la red en la que se encuentren en un momento dado, permitiendo así la movilidad de los nodos.

Funcionamiento

Como se ha dicho anteriormente, un nodo móvil tiene dos direcciones, una permanente (home) y otra dinámica (Care of Address). Además de esta característica de cada nodo, existen dos entidades en Mobile IP:

Home Agent: almacena información de los nodos móviles cuya dirección home pertenezca a la misma red que el agente.

ForeignAgent: almacena información de los nodos móviles cuya dirección home no pertenezca a la misma red que el agente, es decir, de los nodos que han llegado de otras redes.

Cuando un nodo pasa de una red a otra, su Home Agent crea un IP tunnel dirigido a la nueva red en la que se encuentra. Para ello utilizará la nueva dirección Care of Address que habrá obtenido en la nueva red. De este modo, si un host quiere comunicarse con el nodo móvil, enviará los paquetes a su Home IP. Estos paquetes llegarán a la red en la que se encuentra el Home Agent del nodo móvil y este los redirigirá a través del túnel que ha sido creado. Como puede observarse, el cambio de red del nodo móvil pasa totalmente desapercibido para el host emisor.

Cuando el nodo móvil actúa como emisor, este envía los paquetes directamente al destinatario a través del ForeignAgent de la red en la que se encuentre poniendo como dirección IP del emisor su dirección home. Sin embargo esto no es siempre posible, porque puede suceder que el gateway de la red en la que se encuentre en nodo móvil tenga activado el *ingressfiltering*, que fuerza a que los paquetes que salen de su red tengan una IP perteneciente a esa misma red (cosa que el nodo móvil no cumple). En esos casos, el

ForeignAgent utiliza *reverse tunneling* para redirigir los paquetes al Home Agent del nodo móvil, y que este envíe los paquetes al destinatario final.

HIP

Los constantes cambios en la topología de una red y la frecuencia con la que los nodos abandonan la red, hace que el uso de las direcciones IP como identificador sea poco recomendable, ya que las direcciones cambian con frecuencia (un ejemplo claro lo tenemos en la telefonía).

Por ese motivo, es recomendable un nuevo sistema para identificar estos nodos, para que no pierdan o cambien su identidad cuando cambien de dirección IP.

HIP es un protocolo para Internet que permite establecer conexiones seguras entre hosts, y mantener estas conexiones aunque la localización (dirección IP) de los hosts cambie.

Esto se consigue con una capa en la pila de protocolos, en particular entre las capas de red y transporte. Durante el establecimiento de la conexión, los hosts intercambian sus claves públicas y acuerdan una clave secreta de sesión. HIP asume que inicialmente se conoce (o se puede consultar en un DNS) la IP y la identidad del host con el que se quiere establecer la comunicación.

El protocolo

Las direcciones IP, que se diseñaron para establecer una jerarquía con la que poder enrutar correctamente los paquetes, actualmente se utilizan también para identificar los nodos finales.

Este segundo uso tiene el inconveniente de que cuando un equipo cambia su localización en la red (y por tanto cambia de dirección IP) pierde su identidad y pasa a tener una nueva y por ende se pierde la conexión.

El protocolo HIP (Host IdentityProtocol) propone una nueva capa en la pila de protocolos. Este nuevo nivel proporciona autenticación, con independencia de la dirección IP del host. Con esta nueva capa, la función de las direcciones IP pasa a ser únicamente la de enrutar los paquetes.

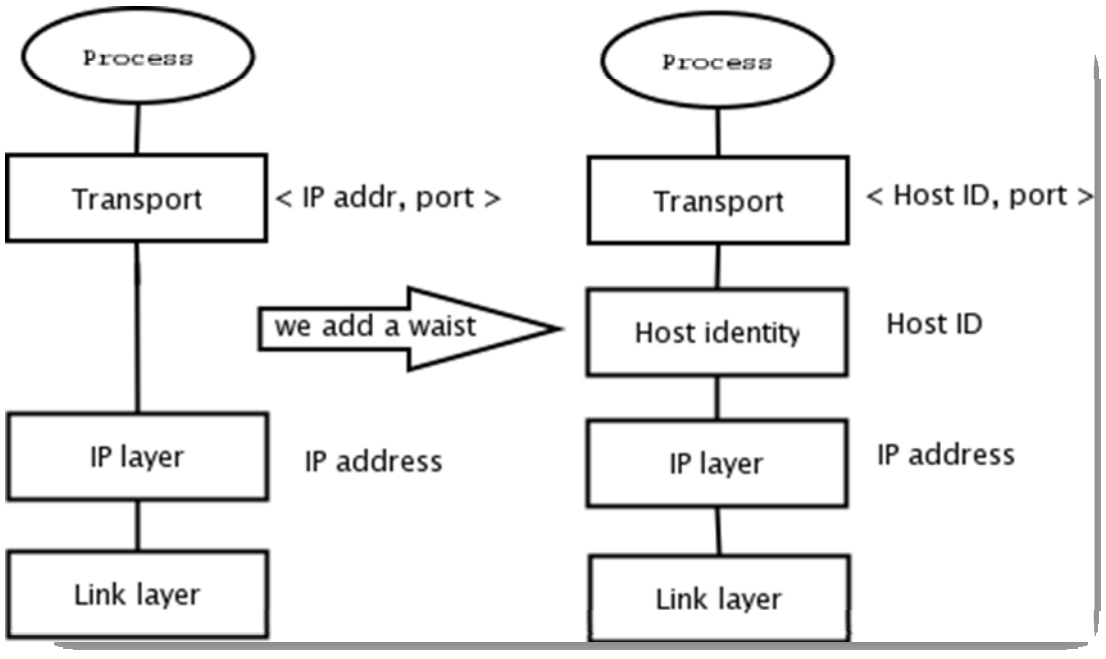


Ilustración 2 : Pila de protocolos con HIP

Esta nueva pila de protocolos se muestra en la siguiente figura:

HIP se encarga de que los usuarios no pierdan la identidad que poseen aunque cambien de ubicación en la red o cambien de dirección IP. Estos cambios son totalmente transparentes para el usuario si utiliza HIP.

Hip introduce un nuevo intercambio criptográfico, el host identity Base Exchange. Este nuevo intercambio permite a los peers establecer SA-s utilizados por IPsec para la creación deuna conexión punto a punto seguro. El tráfico de datos de esta conexión está asegurado mediante los paquetes ESP.En la actualidad, ESP es la mejor manera de proteger la carga en un datagrama. HIP también añade características multi-homing y de movilidad a los dispositivoshabilitados por HIP habilitado.

HIP es, básicamente, un intercambio de claves de ida y vuelta de Diffie-Hellman, un “Base exchange”, y algunos mensajes adicionales.La “Base exchange” se hace para confirmarque dos peers tienen sus propias claves privadas correspondientes a sus propias HI-s, que son las claves públicas. Cuando un host se ha autenticado el “Base exchange” establece dos SA-s paraquela seguridad punto a punto sea encapsulada por ESP.

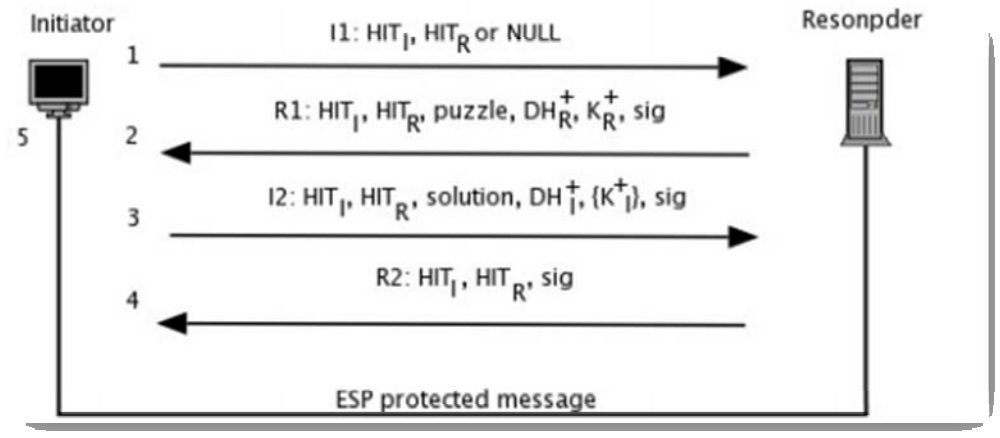


Ilustración 3 : HIP "Base exchange" (intercambio base)

El flujo del “Base exchange” se puede describir enlos siguientes pasos:

Iniciador -> Directorio: buscar respondedor
Iniciador <- Directorio: El respondedor responde con dirección y HI / HIT
I1: Iniciador -> Respondedor (Hola, Este es mi I1, vamos a hablar con HIP)
R1: Respondedor -> Iniciador (Ok, aquí está mi R1,manejar esta cookie HIP)
I2: Iniciador -> Responder (Computando, aquí está mi contador I2)
R2: Responder -> Iniciador (OK. Vamos a terminar HIP con mi R2)
Iniciador -> Responder (Datos protegidos por ESP)
Responder -> Iniciador (Datos protegidos por ESP)

Un nodo es móvil cuando se puede cambiar su ubicacióntopológicaen Internet. Cambiando la ubicación también significa que la dirección IPha cambiado. Un nodo es “multi-homed” cuando setiene más de una conexión simultánea a Internet.El nodo tiene múltiplesrutas para el tráfico entrante y saliente a través de diferentes interfaces. Debido a que los SA-s noestán ligados a direcciones IP, un nodo es capaz de recibir paquetes ESP creados por HIP de cualquier dirección IP. Un nodo puede cambiar suubicación topológicay continuamenteenviar y recibir paquetes desde y hacia sus compañeros

HIP, HIT y LSI

Cada host que implemente HIP tendrá al menos una HI (Host identifier), que es único globalmente. La identidad es el encargado de separar la localización y la identificación. El HI también se puede usar para autentificar las máquinas puesto que es una clave pública.

Para crear una identidad se deberá generar un par de claves, una pública y otra privada. El identificador del host se obtiene a partir de la clave pública. El HI o identidad nunca se usa directamente en internet.Por esto hay que hacer independizar el protocolo HIP del algoritmo de encriptación y conseguir un identificador de longitud fija. A la clave pública se le aplica una función Hash. De este modo se consigue una cadena de longitud fija 128 bits,

denominada HIT (Host IdentityTag), y que aparecerá en la cabecera de los paquetes HIP para identificar al emisor y al receptor.

El demonio HIP que se ejecuta en la máquina, detectaría que se intenta enviar un paquete, y realizaría los pasos necesarios para obtener la dirección IP del host receptor, y así establecer la conexión HIP.

Las direcciones más comúnmente usadas son las direcciones IPv4, de longitud 32 bits. Para poder realizar pruebas con el protocolo HIP, se ha alcanzado una solución de compromiso llamada Local ScopeIdentifier (LSI). LSI es un identificador de 32 bits que se construye directamente a partir de un HIT. Su tamaño le permite ser utilizado en lugar de la dirección IPv4 en aplicaciones y librerías ya existentes.

El demonio HIP se encargará de detectar los paquetes que pretenden enviarse por la red a una LSI en lugar de a una dirección IPv4, obtendrá la dirección IP de la máquina destino y establecerá la conexión HIP.

Convivencia entre IPv4 y HIP

La convivencia entre IPv4 y LSI es resuelta de la siguiente forma. Se asumirá que todas las direcciones IP en el rango 1.0.0.0/8 son en realidad LSI, y los paquetes que intentan enviarse a dichas direcciones tendrán que ser capturados por el demonio HIP, que buscará la dirección IP real del destinatario y establecerá la asociación HIP.

Las direcciones IP comprendidas entre 1.0.0.0 y 2.255.255.255, pese a ser direcciones IP válidas, están reservadas por la IANA (Internet AssignedNumbersAuthority) y actualmente no están en uso. Por tanto, no habrá ningún conflicto entre los paquetes que deben ser enrutados hacia Internet y los paquetes con destino 1.0.0.0/8 que deben ser manejados por el demonio HIP.

El protocolo HIP no especifica cómo construir el LSI. La implementación del protocolo HIP con la que se ha trabajado en este proyecto lo construye copiando los 24 bits de menor peso del HIT en los 24 bits de menor peso del LSI. Los 8 bits de mayor peso del LSI serán siempre 00000001.

El demonio HIP intercepta los paquetes IP con una dirección destino en el rango 1.0.0.0/8. A continuación, busca concordancias en el fichero local de identidades conocidas. En la implementación OpenHIP, este fichero se ubica en /usr/local/etc/hip/known_host_identities.xml.

```
<host_identity alg="RSA" alg_id="5" length="128">
  <name>un-dominio.com-1024</name>
  <addr>81.124.2.201</addr>
  <HIT>2001:19:cae1:2f79:8f5b:d8c6:4a2f:c1af</HIT>
  <LSI>1.23.55.233</LSI>
</host_identity>
```

Ilustración 4 : identificación de un host a través de HIP

Este fichero puede contener, para cada identidad HIP conocida (<host_identity>), su nombre de dominio (si tiene), su dirección IP, su HIT y su LSI. Si no se conoce el LSI, éste se calculará de forma directa a partir del HIT.

Cuando el demonio HIP intercepta un paquete IP con dirección IP destino 1.0.0.0/8, busca esa dirección en los campos LSI del fichero local de identidades conocidas (known_host_identities.xml en OpenHIP). Si aparece junto a su HIT y dirección IP real, se intenta establecer la asociación HIP. Una vez establecida la asociación HIP, se enviará el paquete interceptado. Si aparece en el fichero el LSI pero no junto a su HIT, se intentará establecer la asociación en modo oportunista. El modo oportunista es aquel que intenta establecer una asociación HIP con un host cuya identidad se desconoce. El hostremoto podrá aceptar o rechazar las solicitudes oportunistas.

VPN

Virtual privatenetwork (red privada virtual). Es un sistema que extiende una red privada sobre una red pública, como puede ser Internet. Tiene un gran número de usos, como conectar dos entidades, hacer que un usuario desde su casa trabaje como si estuviese en la oficina, o acceder a un ordenador remotamente.

Para poder llevar a cabo la conexión, es necesaria la autentificación y autorización del usuario y el equipo sobre el servidor de vpn. Los requerimientos para esta conexión son:

Identificación del usuario
Codificación de los datos (mediante algoritmos cifrados)
Actualización de las claves por parte del servidor VPN para los usuarios

Tipos de Vpn

1. *VPN de acceso remoto:* Un usuario se conecta a la red privada desde fuera de ella, y tiene acceso a los recursos como si estuviera dentro de la red.
2. *VPN punto a punto:* Conectar oficinas remotas con la sede central. El servidor VPN acepta las conexiones, estableciendo un túnel VPN. Es más común tunneling.
3. *Tunneling:* Encapsular un protocolo de red sobre otro, creando un túnel dentro de una red de computadores. El túnel se implementa incluyendo una PDU (trama de datos) encapsulado dentro de otro, de tal forma que no sea necesaria una interpretación intermedia de la PDU encapsulada (sólo se interpreta entre los extremos del túnel). Es decir los paquetes viajan por el túnel, pero los nodos intermedios no pueden ver su contenido. Un uso común puede ser la *IPMóvil*. Cuando un nodo móvil no está en su red base, su home-agent (ordenador ficticio que se está en la red local) captura el tráfico dirigido al nodo-móvil, y se lo redirige, usando tunneling, pues es necesario que los paquetes tengan la estructura y datos originales (direcciones IP origen y destino, puertos, ….) cuando sea recibida por el nodo-móvil.
4. *VPN over LAN:* Versión de acceso remoto. En vez de comunicarse por Internet, utiliza la propia red de área local de la empresa. Sirve para aislar zonas y servicios de la red interna. El servidor a proteger se aísla detrás del equipo VPN, que pide autentificación, y sólo los que tengan autorización pueden entrar. En Wifi, los túneles ipsec o ssl, además de restricciones tradicionales (como wep, o wap, …) agregan las credenciales del túnel VPN.

Implementación

El protocolo estándar es el de IPSEC, pero existen otros (pptp, l2f, ssh,…). Con ssh están apareciendo varios productos actualmente. Según el origen del servicio VPN, podemos distinguir dos tipos claros:

- *Hardware:* Ofrecen mayor rendimiento y fiabilidad de configuración, aunque es menos flexible. Tenemos una gran variedad de ejemplos, como SonicWall, Cisco, Linsys, Nokia, D-Link,….
- *Software:* Más configurables a las necesidades y más fácil de adaptarlos, pero ofrecen un menor rendimiento. Además, pueden presentar problemas añadidos con el sistema operativo por motivos de seguridad. Existen soluciones tanto en Windows como en Linux, y también de código abierto, como openssh, openvpn,….

Tipos de conexión:

- *Conexión de acceso remoto*: Un cliente se conecta a una red privada. El cliente se autentifica contra el servidor de acceso remoto, y el servidor contra el cliente.
- *Conexión VPN router a router*: La conexión es realizada por un router, y éste a su vez se conecta a una red privad. Los dos routers se autentifican. Los paquetes enviados desde cualquier router no se originan en los routers, sino en los pc's conectados a éstos.
- *Conexión VPN firewall a firewall*: Análogo a la conexión VPN router a router.

Para desarrollar la tecnología VPN en el proyecto se ha recurrido a la herramienta **openvpn** (http://openvpn.net/), una solución de conectividad basada en software (SSL), de la que ya se habló anteriormente.

Existe una red central (servidor) y unas oficinas remotas (clientes). Se pueden dar distintos permisos a distintos usuarios, los cuales se tienen que autentificar contra el servidor.

Cada oficina tiene su propia LAN, y un túnel distinto. El servidor VPN hace de pasarela; ahora queda como una red local, ya que él captura el tráfico que va hacia sus clientes. Los clientes están conectados por una red virtual proporcionada por el servidor.

Para la configuración de openvpn en el proyecto, se decidió por utilizar un bridge en el servidor, y un tap para poder conectarse a él. Un *bridge* (o puente) es un dispositivo de interconexión de redes de ordenadores, que actúa en la capa de datos. Sirve para conectar dos segmentos de red como si fueran una única. Un *tap* es una interfaz que se crea para hacerse ver en una red virtual, para que la máquina con el tap se pueda ver en la red en la que se autentifica como si estuviera en ella. Así, el bridge se levantará en el servidor openvpn, y el tap estará en el cliente openvpn cada vez que inicie el proceso y se conecte con el servidor, por lo que, aparentemente, el pc está en la red, aunque su ubicación física no esté en esa red.

FASES DE PROYECTO

Introducción

Desde el momento en el que se comenzó con el proyecto, se intentó establecer un plan de desarrollo para este, para que el proyecto tuviera una organización temporal que permitiera establecer un alcance para el mismo.

Primera fase: Formación

La situación del grupo en cuanto a conocimientos referentes a los conceptos que se iban a manejar en el proyecto era bastante precaria, puesto que nunca antes los componentes del grupo habían tenido que tratar con temas relacionados con las redes a pesar de haber cursado asignaturas en con estos conceptos. Se podría decir que se tenían algunos conocimientos teóricos, pero a todas luces inferiores a los necesarios para conseguir resultados satisfactorios.

La primera fase del proyecto, por tanto, estaba clara y no era otra que la formación en temas relacionados con las redes en general, no teniendo necesariamente que ver con los protocolos concretos que se utilizarían en el proyecto. No se preveía que esta fase durase demasiado tiempo, ya que entre los tres componentes del grupo resultaría bastante sencillo intercambiar los conocimientos que ya se tenían. Se puede decir que en una o dos semanas el grupo estaría dispuesto en términos de conocimientos teóricos generales sobre redes.

Segunda fase: Primeras pruebas generales

Una vez que el grupo tuviese unas primeras bases para comenzar a realizar pequeñas pruebas con la creación de infraestructuras de red estas fueron diseñadas para comenzar a tener los primeros contactos reales con los conocimientos teóricos obtenidos en la primera fase. Además de esto, las primeras pruebas sirvieron para ir estableciendo los parámetros en los que se basaría el entorno de trabajo y de pruebas.

Las pruebas mencionadas se basaron simplemente en conectar unos pocos nodos en una o varias redes y conseguir que los ordenadores de dicha red se conectaran correctamente entre sí mediante el encaminamiento simple de paquetes. Como se ha dicho también, se estableció el sistema operativo de los ordenadores involucrados en las pruebas, y el sistema de virtualización que se utilizaría para simular los ordenadores. Estos puntos se tratan más en detalle en capítulos posteriores de la memoria.

Tercera fase: investigación de protocolos

Después de conseguir hacer funcionar las primeras infraestructuras de red de la fase anterior, se comenzó a investigar los protocolos que se utilizarían en el proyecto. Dado que en un principio estos eran tres, y que el grupo tenía tres componentes, se repartió un protocolo para cada componente.

Esto permitió un estudio en paralelo de los tres protocolos, lo cual ayudó a avanzar más rápidamente en las fases del estudio de las tecnologías.

Cada componente del grupo se comprometió a estudiar el protocolo que le fue asignado, entenderlo y escribir un pequeño documento de explicación para poder después transferirlo al resto del grupo.

Esta fase, lógicamente, también comprendió una fase de comunicación entre los componentes del grupo, para que todo el grupo estuvieran al tanto de los pormenores de los cada uno de los protocolos.

Cuarta fase: instalación y pruebas con los protocolos

Llegado a este punto el grupo tiene una base tanto en conocimientos de redes como de los protocolos objeto de estudio, y también se han realizado las primeras pruebas con el entorno de trabajo, por lo que el grupo está preparado para poder avanzar en las pruebas reales. Para poder hacer esto, es necesaria una fase de instalación de los protocolos en las máquinas virtuales. Aprovechando que cada componente del grupo había estudiado un protocolo, se decidió mantener la misma organización para las tareas de instalación. De esta forma, el grupo avanzó con las instalaciones en paralelo.

Esta fase es la que más problemas dio en el proyecto, ya que el grupo tuvo dificultades para conseguir hacer funcionar las soluciones. Fue en esta fase en la que se decidió desechar uno de los protocolos, Mobile IP, porque se vio que el protocolo estaba bastante abandonado para la versión 4 del estándar IP. Finalmente, se consiguió que las otras dos opciones funcionaran por lo que se pasó a la siguiente fase, el diseño de las pruebas.

Quinta fase: diseño de las pruebas

En esta fase, se hizo una reestructuración del grupo de trabajo, ya que al haber dejado de lado Mobile IP un componente del grupo quedó sin protocolo que estudiar. En este punto, se decidió que dos componentes del grupo se pusieran a trabajar con Open VPN, puesto que en aquel momento era la parte más retrasada del proyecto.

En esta fase se decidió cuáles serían las pruebas a realizar para llevar a cabo la evaluación objetivo del proyecto. Este diseño de las pruebas sería utilizado por ambos protocolos, ya que es necesario que las pruebas sean iguales para ver que protocolo se comporta de mejor forma.

El diseño de las pruebas estableció que se harían testeos con diferentes tecnologías para probar los protocolos en diferentes escenarios. Estas pruebas están detalladas más adelante, por lo que en este apartado no se especificará nada más sobre estas.

Sexta fase: ejecución de pruebas y recogida de resultados

Esta fase simplemente sirvió para ejecutar las pruebas diseñadas en la fase anterior. Las pruebas se separaron entre los componentes del grupo para que cada componente pudiera avanzar de forma paralela.

Una vez realizados los testeos por separado, llegaría el momento de juntar todos los datos obtenidos para tener todas las pruebas en como un todo.

Séptima fase: Interpretación de resultados y conclusiones

La última fase es en la que se sacaron las conclusiones finales de los resultados obtenidos en las pruebas. Las conclusiones fueron simplemente un resultado de la interpretación de toda la información de la que se disponía hasta el momento.

Octava fase: Fin del desarrollo de la memoria y entrega

Después de todo el desarrollo y una vez alcanzadas las conclusiones, la última parte del proyecto fue la dedicada al cierre de este. Dentro de esta fase se finalizó la memoria del proyecto y se realizó todo el proceso de entrega del proyecto. Este proceso comprende desde la entrega del borrador de la memoria hasta la presentación pública del proyecto. Todo este proceso fue realizado en el mes de septiembre de 2011.

DESARROLLO

Introducción

En el siguiente apartado de la memoria se explicaran los pasos seguidos para desarrollar el proyecto. Se hablará de cuál ha sido la infraestructura utilizada, así como el proceso de obtención de dicha estructura. Será también en este apartado donde se mostraran los resultados obtenidos en las pruebas. Una vez presentados los mencionados resultados se pasará a sacar las conclusiones que de ellos se deriven.

Entorno de trabajo

A la hora de la realización de las pruebas, estas pueden separarse en dos grupos: las realizadas en un entorno virtualizado y las realizadas con máquinas reales. Desde un principio se decidió que las pruebas se harían mediante máquinas virtuales para no tener que depender de tener ordenadores físicos, ya que no siempre es posible tener todos los recursos necesarios. Sin embargo, el grupo de trabajo decidió a medida que el proyecto avanzaba que sería interesante hacer estas pruebas en un entorno real, es decir, utilizando máquinas físicas.

Una vez tomada esta decisión, se llegó a la conclusión de que lo más conveniente sería hacer exactamente las mismas pruebas tanto en el entorno virtualizado como en el real. De esta forma sería posible tener una visión más realista del funcionamiento de los protocolos, y al mismo tiempo ver cuáles son las inexactitudes introducidas en los resultados por el uso de un entorno virtualizado.

En los siguientes dos apartados se describirán los dos entornos utilizados para las pruebas.

Entorno virtualizado

La evaluación de las tecnologías presentadas anteriormente se fundamenta en el uso de virtualización para simular diferentes máquinas físicas, ya que de lo contrario serían necesarias demasiadas máquinas físicas para ir haciendo las pruebas, cosa que no siempre estaba al alcance del grupo de trabajo.

Además de la razón esgrimida anteriormente, también está el hecho de que este es un proyecto de movilidad IP, y para hacer las pruebas se necesitan máquinas que cambien de una red a otra, por lo que si las pruebas fueran realizadas con máquinas físicas, sería necesario ir caminando para pasar de una red a otra y ver el comportamiento del sistema. Obviamente este modo de trabajo no era viable, por lo que el uso de máquinas virtuales era el mejor camino a seguir.

El grupo de trabajo escogió para virtualizar máquinas el software desarrollado por Sun Microsystems (ahora propiedad de Oracle) VirtualBox.

Utilizando VirtualBox se puede crear máquinas virtuales, es decir, máquinas que corren bajo un ordenador físico y que disponen de su propio ambiente virtual. Los sistemas

virtualizados dentro de una máquina física se llaman “máquinas invitadas” y al ordenador físico se le conoce como “máquina anfitriona”.

A pesar de que VirtualBox soporta multitud de sistemas operativos para la máquina anfitriona, el grupo de trabajo ha funcionado con el sistema operativo GNU/Linux, y concretamente con la distribución Linux Ubuntu 10.10. De la misma manera que con el anfitrión, las máquinas invitadas se han ejecutado bajo Ubuntu 10.10.

El hardware del que disponen las máquinas virtualizadas es el siguiente:

- *Procesador:* utiliza el mismo que el de la máquina anfitriona, con la salvedad de que se le asigna un único núcleo.
- *Disco duro:* Se le asigna a cada máquina virtual un disco duro de 8GB.
- *Memoria RAM:* 512MB para cada una.
- *Tarjeta gráfica:* Utiliza la misma que la máquina anfitriona, pero sólo tiene 12MB de memoria para la pantalla.
- *Tarjeta de Red:* Puede elegir uno de los 6 tipos disponibles para cada adaptador de red que asociemos a la máquina virtual, pero las máquinas con las que se ha desarrollado el proyecto tienen Intel PRO/1000 MT Desktop (82540EM)

Un componente básico para las pruebas son las interfaces de red virtuales. En cada una de las máquinas virtuales es posible simular hasta ocho interfaces de red, aunque solo es posible configurar cuatro desde el interfaz de usuario. En caso de necesitar más de cuatro interfaces de red, sería necesario configurarlas utilizando la línea de comandos. Sin embargo, para este proyecto no han sido necesarias más de cuatro, por lo que con el interfaz de usuario proporcionado por VirtualBox ha sido suficiente para crear la infraestructura necesaria.

Además de configurar un número concreto de interfaces de red, también existe la posibilidad de especificar al sistema de virtualización la forma en la que cada interfaz de una máquina virtual se conecta a la red a la que esté conectada. A continuación se presenta una breve explicación de los diferentes modos de red que ofrece VirtualBox:

Sin unir

En este modo, VirtualBox notifica al sistema invitado que existe una tarjeta de red, pero que no hay conexión -- como si el cable Ethernet no estuviera conectado al puerto. De esta forma es posible “tirar” del cable Ethernet virtual e interrumpir la conexión, lo cual puede ser útil para informar al sistema operativo invitado de que no hay ninguna conexión activa y forzar una reconfiguración.

Traducción de Direcciones de Red (NAT)

Si todo lo que se necesita es navegar por internet desde el sistema operativo invitado, este modo es suficiente. El sistema de virtualización envía el tráfico a través de la tarjeta de red del ordenador físico de forma que es posible alcanzar internet desde el invitado.

Bridged networking

Este caso es útil para necesidades más avanzadas como simulaciones de red y ejecuciones de servidores en el invitado. Cuando está activado, VirtualBox se conecta a una de las tarjetas de red instaladas y extrae los paquetes directamente, eludiendo la pila de red del sistema anfitrión.

Internal networking

Este modo puede ser utilizado para crear redes basadas en software que sean visibles para las máquinas virtuales seleccionadas, pero no para las aplicaciones que se ejecutan en el anfitrión o en otras máquinas virtuales no seleccionadas para ver dicha red.

Host-only networking

Si se escoge esta opción se crea una red que contiene al invitado y un grupo de invitados, sin la necesidad de la interfaz de red de la máquina anfitriona. En lugar de ello, se crea en el anfitrión una interfaz de red virtual (similar a la interfaz loopback), proporcionando conectividad entre la máquina física y las virtuales.

Para hacer las pruebas, se han necesitado diferentes redes de ordenadores, para poder efectuar los cambios de una a otra. Para cubrir esta necesidad, se ha hecho uso de host-only network. Además, para tener acceso a internet desde las máquinas virtuales de forma fácil se ha utilizado NAT, ya que de esta forma se obtiene acceso a través del ordenador anfitrión.

Todo lo mencionado en este apartado es suficiente para poner una infraestructura mínima en funcionamiento, a partir de la cual comenzar a instalar las tecnologías a evaluar y a hacer las pruebas pertinentes.

Entorno físico

Una vez realizadas las pruebas en el entorno se virtualización, se paso a hacer estas mismas pruebas utilizando ordenadores reales. En la mayoría de las pruebas realizadas fueron necesarias tres máquinas y dos redes. La base de esta infraestructura fue un ordenador de sobremesa al que se le instalaron dos interfaces de red para hacer posible que se conectara a sendas redes.

La gestión de las redes se delegó en dos routers, de forma que las máquinas tuvieran la posibilidad de conectarse a una u otra red mediante cable Ethernet o WiFi.

Para las pruebas, el ordenador de sobremesa actuaría de router entre las dos redes. La razón de no usar los routers de los que se disponía fue la necesidad de poder configurar el router para que funcionase específicamente como requerían las pruebas, y se llegó a la conclusión que sería más sencillo hacer esto en un ordenador que en uno de los routers. Por tanto, la única función de los routers reales sería la de poner a disposición de las máquinas las dos redes mencionadas con anterioridad.

A la hora de simular un cambio de red utilizando las máquinas reales, se ha jugado con desconectarse de una red y conectarse a la otra, tanto con la línea de comandos como utilizando el entorno gráfico del sistema operativo.

Las siguientes listas, especifican las máquinas utilizadas para las pruebas y las especificaciones de todas ellas:

- **Dell Studio 17**
  - Procesador Intel Core 2 Duo P8400
  - Adaptador de red inalámbrico Intel Wifi Link 5100
  - Adaptador Ethernet Broadcom NetLink BCM5784M Gigabit Ethernet PCIe
  - Tarjeta gráfica ATI Mobility Radeon HD 3650
  - Memoria RAM de 4 GB
  - Disco duro de 320 GB
- **Dell Inspiron 1750**
  - Procesador Intel Core 2 Duo P8600 (2.40GHz)
  - Dell Wireless 1397 802.11b/g half mini-card
  - Adaptador Ethernet Broadcom Corporation BCM4312 802.11b/g LP-PHY

    - Tarjeta gráfica Mobility Radeon HD 4300 Series
    - Memoria RAM de 4 GB
    - Disco duro de 320 GB

- **Lenovo G550**
    - Intel Pentium III Xeon Dual-Core, T4500, 2300 MHz
    - Microsoft Virtual WiFi Miniport Adapter
    - Adaptador de red Broadcom 802.11g
    - Tarjeta gráfica Mobile Intel(R) 4 Series Express Chipset Family
    - Memoria RAM de 4 GB
    - Disco duro de 500 GB
- **Router SpeedTouch**
    - Speed touch 585
    - Tipo de Interfaz: 802.11b/g
    - Modo de seguridad: WPA-PSK
    - Velocidad exacta: 54 Mbps
- **Router Jazztel**
    - Router Jazztel 96328A-1241N
    - Tipo de Interfaz: 802.11b/g
    - Modo de seguridad: WPA-PSK
    - Velocidad exacta: 54 Mbps

Software utilizado

La evaluación de los protocolos OpenVPN y HIP se ha llevado a cabo con las siguientes implementaciones y sus correspondientes versiones:

- OpenVPN 2.1.3
- HIP 7.0

A pesar de que no se pudo desarrollar prueba laguna con el protocolo Mobile IP, cabe mencionar que sí se estudió en términos teóricos y que se intentó poner en marcha en la infraestructura de pruebas. La implementación de Mobile IP y su versión ha sido la siguiente:

- Dynamics Mobile IP 0.8.1

Además de las implementaciones de los estándares objeto de estudio, una herramienta que a resultado de vital importancia ha sido Wireshark. Esta herramienta es un analizador de protocolos de red que permite estudiar todo el tráfico generado en una red de computadores. Mediante el uso de Wireshark, es posible capturar todos los paquetes intercambiados entre diferentes máquinas, y poder estudiar a posteriori dicho tráfico.

Para terminar con el apartado de software utilizado, hay que mencionar todo lo relacionado con las herramientas de desarrollo utilizadas para generar aplicaciones durante el proyecto. Estas herramientas han sido las siguientes:

- ***NetBeans***

Entorno de desarrollo de orientado principalmente al desarrollo Java. Es el entorno que se ha utilizado cuando ha sido necesario el desarrollo de aplicaciones mencionadas en el siguiente apartado.

- ***JFreeChart***

Librería para el lenguaje de programación Java que proporciona generación de gráficas de todo tipo a partir de ciertos datos de entrada. Se ha utilizado en las aplicaciones de las que se habla en el anterior punto para generar las gráficas presentes en esta memoria.

- ***Shell***

Se han desarrollado algunos scripts que han permitido el manejo de las redes a la hora de hacer las pruebas, dado que es un intérprete de comandos que permite introducir diferentes comandos desde un fichero.

- ***Matlab***

Matlab es un software matemático que ofrece un entorno de desarrollo integrado con un lenguaje de programación propio. Se ha hecho uso de Matlab para generar las gráficas comparativas que aparecen en este documento en la sección destinada a comparaciones.

Software desarrollado

Una parte importante de las pruebas ha sido la obtención de información y el uso de esta para extraer datos entendibles. Solo así ha sido posible sacar las conclusiones que se derivan de las pruebas realizadas.

A menudo, la información obtenida era muy extensa y en un formato muy poco legible para el humano. Debido a esto el grupo de trabajo desarrolló algunas pequeñas aplicaciones con el fin de que fueran estas las que generaran información legible y recursos gráficos de forma rápida.

La principal tarea que se delegó en estos programas fue la de recibir una información de tráfico de red como entrada y generar información y gráficas referentes a ella. Con estos resultados el grupo de trabajo tenía información con la que llegar a las conclusiones descritas más adelante en este documento. En las siguientes líneas se describe brevemente cada aplicación:

✓ *PacketReader:* Esta aplicación fue la primera que se desarrolló. Su objetivo era capturar una lista de paquetes intercambiados en una red, clasificarlos y extraer de la lista los paquetes que interesaban. Una vez que se conseguía la lista de paquetes relevantes para las mediciones, la aplicación sacaba métricas a partir de dicha lista. Entre otras cosas, daba como resultado las peticiones totales enviadas y las respuestas recibidas y la media de tiempos entre una petición y su correspondiente respuesta. Además de esto, la aplicación era capaz de generar gráficas utilizando los datos de entrada. Varias de las gráficas que se encuentran en esta memoria han sido generadas por PacketReader. Esta aplicación ha sido útil para sacar los resultados de las pruebas con muchas peticiones, como por ejemplo las pruebas con ping o las peticiones HTTP.

✓ *DownloadTimeComparer:* Esta aplicación es menos compleja que la anterior. Se ha utilizado para comparar tiempos de descarga o de transferencia. Lo único que hacía era recibir algunos tiempos como entrada y generar gráficas que, de nuevo, están presentes en esta memoria. Este programa ha sido utilizado, por ejemplo, para generar gráficas de las pruebas de servidor web y la descarga de un fichero grande.

INFRAESTRUCTURA UTILIZADA PARA LAS PRUEBAS

Antes de comenzar a tomar contacto con las tecnologías a evaluar en el proyecto, el grupo de trabajo y el profesorado acordaron crear en primer lugar una infraestructura mínima con la que hacer las primeras pruebas de red.

Primera red con dos hosts y un router

El primer objetivo de esta fase fue poner en funcionamiento una infraestructura de red en la que hubiera dos redes interconectadas mediante un router (o un PC que hiciera la función de router) y que una red fuera alcanzable desde la otra y viceversa. La razón para crear la infraestructura con la topología mencionada es que en las tres tecnologías que se iban a evaluar iba a ser necesaria la existencia de dos redes, con las cuales simular el cambio de red de los nodos móviles.

Por lo tanto, se procedió a crear dicha infraestructura, en la que existían en un principio tres máquinas: un router y dos hosts. El objetivo de esta infraestructura era tener dos host (servidor y cliente) en dos redes diferentes, conectados a través de un solo router, y que hubiese conexión entre ellos.

La siguiente figura muestra la organización de la infraestructura configurada.

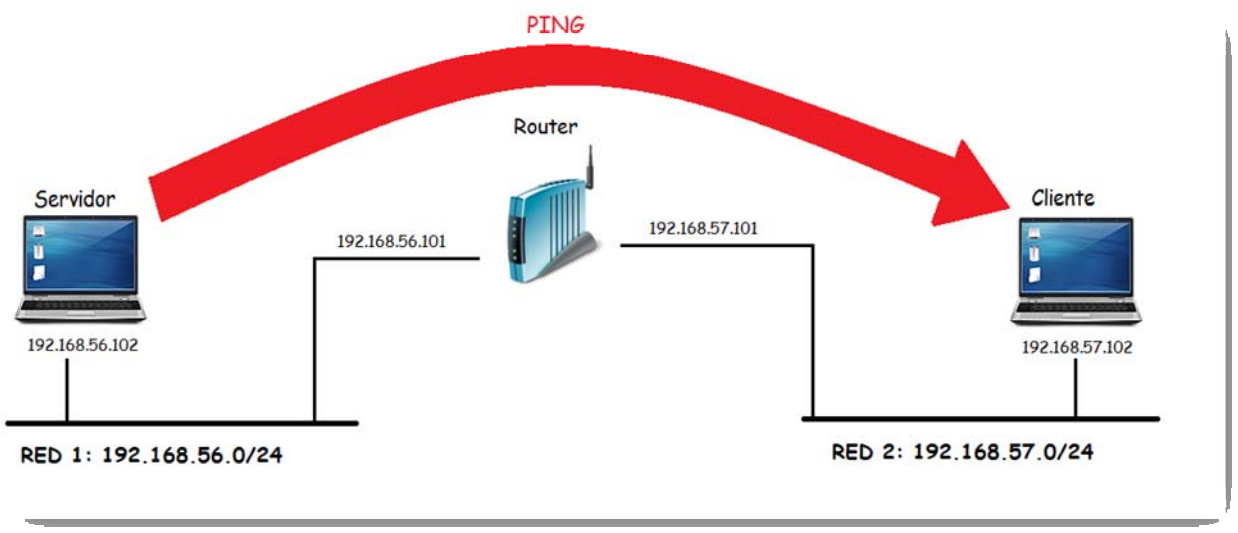


Ilustración 5 : Primera infraestructura virtual

Una vez puestas en marcha todas las máquinas fue necesario configurar la máquina router para que funcionara como tal, ya que inicialmente no había conexión entre el servidor y el cliente. En este caso, solamente era necesario que el router tuviera activado el IP Forwarding. De este modo, la máquina router es capaz de pasar los paquetes de una interfaz a otra, de forma que interconecta las dos redes, y la conexión entre el cliente y el servidor queda establecida.

Para finalizar con la configuración, cada uno de los hosts se configuro para que su puerta de enlace por defecto fuera la dirección ip del router que estaba conectada a su misma red.

Segunda red con dos routers creando dos redes distintas

Una vez conseguida la infraestructura anteriormente especificada y verificada la conectividad entre todas las máquinas, se probó todo el sistema poniendo un servidor FTP para que la otra máquina accediera a él y verificar el funcionamiento de la red. El objetivo de esto fue establecer un entorno con el cual tener una base para hacer pruebas una vez que las tecnologías de movilidad fueran operativas.

Después de conseguir poner en marcha el FTP, se pasó a configurar una red con un pequeño grado más de complejidad. En este caso, los dos hosts que están en los dos extremos tienen dos routers entre medias. De esta forma, los routers deben tener configuradas las rutas a las redes que les quedan más lejos y tener habilitado el IP Forwarding.

En la figura siguiente se muestra gráficamente la organización de la infraestructura.

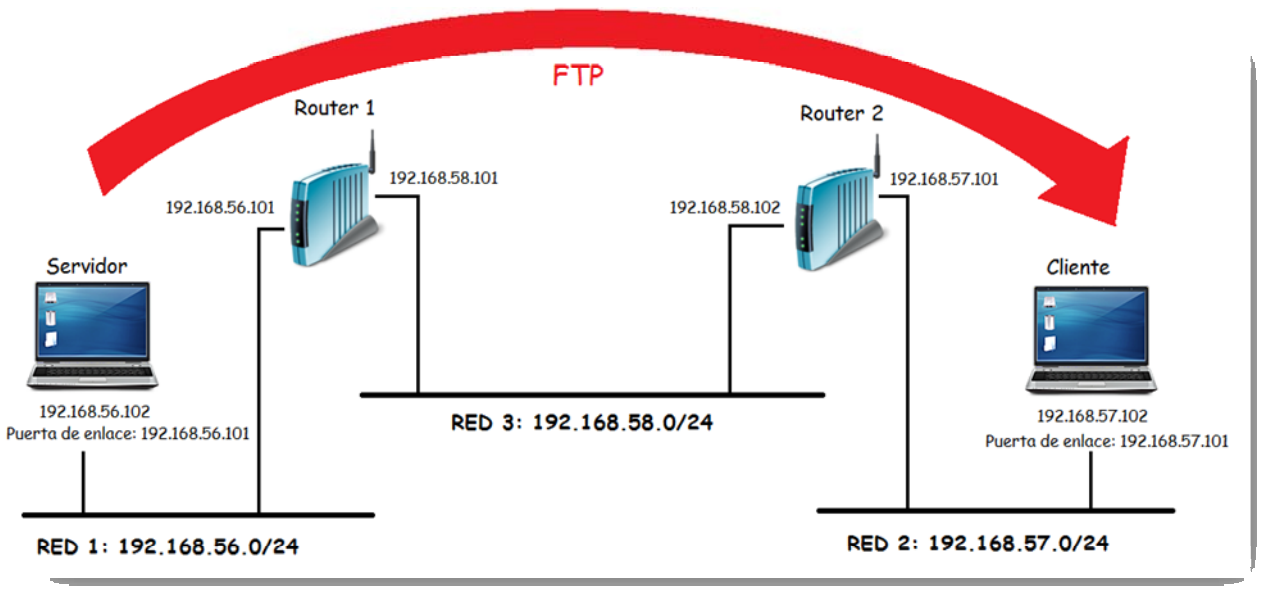


Ilustración 6 : Segunda estructura virtual

Como se puede observar en la imagen, el servidor de ficheros ftp está en la red 1, y el cliente que cogerá datos del servidor se encuentra en la red 2. Para poder llegar a comunicar esas dos máquinas, se crea una red intermedia, la red 3. Para conectar la red de cada máquina con esta red intermedia, se utilizarán dos routers (en nuestro caso, dos máquinas de Ubuntu

cuya función será hacer de router), que tendrán dos direcciones IP, una de cada red particular que se quiere unir, y otra a la red 3, la red común que tiene entre ellos. Como ocurría en la red 1 del ejemplo anterior, las máquinas que ejercen de router tienen que tener habilitado el IP Forwarding para poder pasar los paquetes que reciban de una interfaz a otra.

Llegados a este punto el grupo de trabajo se dividió para abarcar las tres tecnologías objeto de estudio, asignando a cada miembro del grupo un protocolo de movilidad. En los siguientes apartados se hablará más concretamente de cada una de las tecnologías estudiadas y de la infraestructura necesitada para cada una de ellas.

Tercera red: estructura con el cambio del servidor de red

Por último, y una vez estudiadas las estructuras en las máquinas virtuales, y teniendo su configuración adecuada, quedaba montar la estructura con la que se harían las pruebas, tanto a nivel virtual como a nivel real.

Esta estructura consistía en tener dos redes (suministradas por dos routers en el caso de la red física). Inicialmente, tanto el servidor como el cliente estarían en la misma red, e iniciarán la comunicación entre ellos.

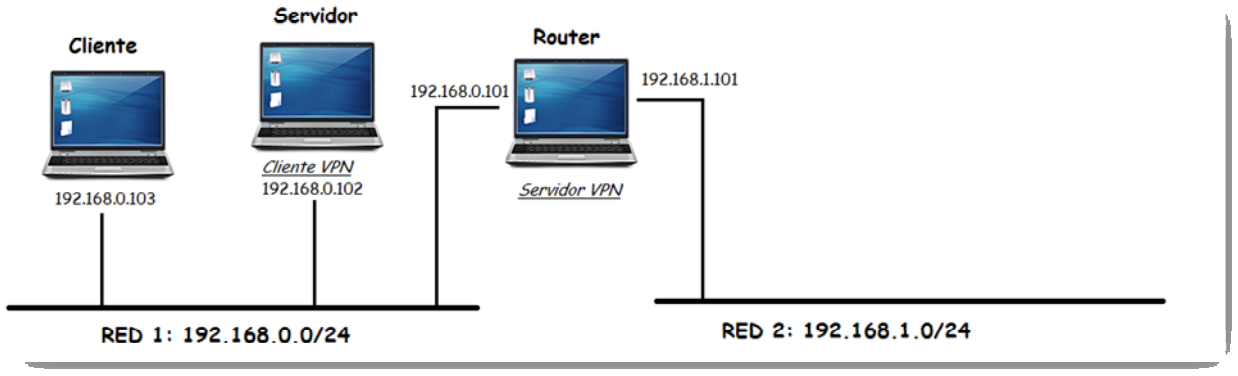


Ilustración 7 : Estructura real en local

En un momento dado, se decide migrar el servidor a la otra red, sin desactivar la conexión. Ésta se verá afectada durante un tiempo, y una vez que se active una de las técnicas de movilidad IP de servidores que estamos estudiando, retomará la conexión, al identificar que el servidor, que se encuentra en la red2, es el mismo con el que inició la conexión en la red1, y ésta se retomará.

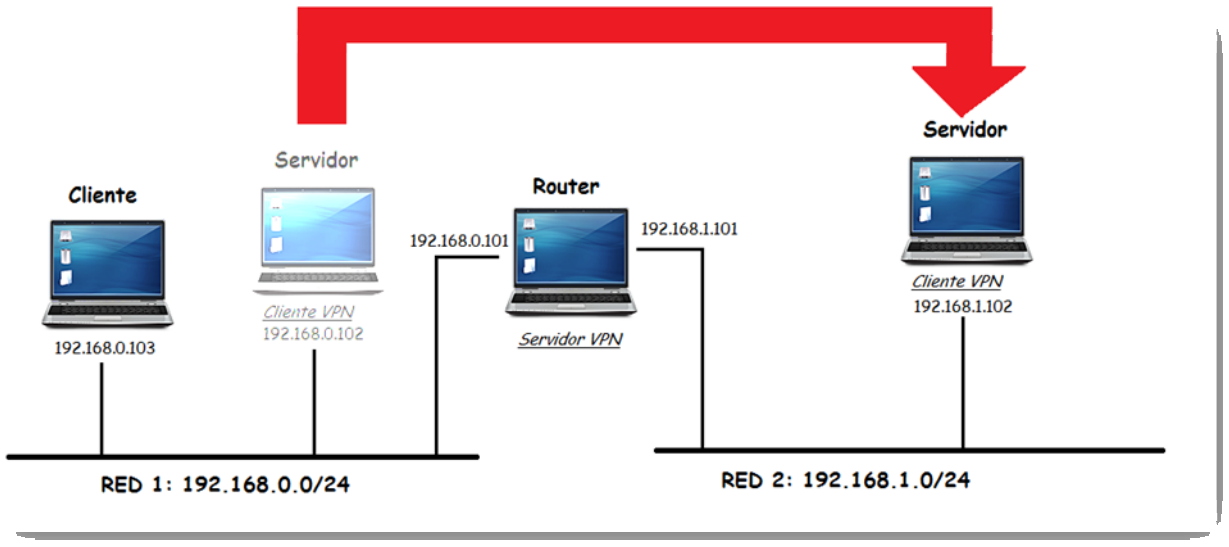


Ilustración 8 : Estructura real con servidor en red pública

Ésta será la estructura que se utilizará tanto para la parte real como para la parte virtual, con la salvedad que en la parte real necesitamos dos routers que ejerzan la función de punto de acceso, y creen las redes 1 y 2.

PRUEBAS A REALIZAR

Una vez elegido el objetivo del proyecto, analizada la problemática de la movilidad IP de los servidores, y elegidas las tecnologías a estudiar, quedaba por elegir las pruebas a realizar para este estudio y comparación.

Entre la gran multitud de protocolos, sistemas de copias y demás pruebas que se pueden hacer para esta actividad, se eligieron las que se consideraron más representativas para el estudio de las redes, más conocidas, más estandarizadas y las que pudieran arrojar datos más claros. Las pruebas elegidas fueron:

Ping

Ping es la herramienta de diagnóstico por excelencia. Comprueba el estado de la conexión en una red TCP/IP entre dos host, mediante el envío de paquetes ICMP. Para ello, el host que realiza la petición de ping envía una solicitud (reply), y el host destino, una vez que la ha recibido, envía a la primera máquina un paquete de respuesta (request). Gracias a éstos envíos de paquetes y a sus mediciones, se puede diagnosticar el estado de la red, midiendo la velocidad y la calidad de la misma.

El mensaje que se envía se encapsula dentro de un paquete (paquete ICMP), en el que se pueden distinguir dos partes muy bien diferenciadas: la cabecera, que contiene los datos de la capa de red, y el subpaquete ICMP (o carga ICMP), que contiene los datos de control. Si se quiere enviar un paquete de mayor tamaño, lo único que habrá que hacer es añadir datos a este subpaquete ICMP.

Cada cierto tiempo, el host fuente emitirá la solicitud de eco, enviando un paquete a la máquina destino. Esta, cuando reciba la solicitud, enviará una respuesta de eco a la máquina fuente, que capturará la respuesta, y calculará el tiempo que se tardó entre la solicitud de envío y el momento de la recepción de la respuesta.

Ping ofrece la posibilidad de modificar las características del envío de paquetes, tanto en tamaño como en frecuencia. Así, para obtener mayor diversidad de resultados y que las pruebas fueran más concluyentes, se eligieron cuatro pruebas distintas con la herramienta ping, que se detallan a continuación:

Ping básico:

ping 192.168.57.102

Es el envío de paquetes por defecto de la herramienta ping. Al no especificar ningún parámetro, usará los valores de configuración que están definidos por defecto, que son: tamaño de paquete: 64 bytes; frecuencia de envío: 1 paquete por segundo.

Ping con un tamaño de 1000 bytes por paquete

ping 192.168.57.102 –s 992

Ahora, con la finalidad de conocer el comportamiento de los protocolos con un tamaño mayor de los paquetes enviados, se utiliza éste comando, con el parámetro –s 992. El parámetro –s se refiere a size (tamaño), y el motivo de poner 992 es que con ese número se define el tamaño del cuerpo (carga ICMP); no se incluyen los 8 bytes que lleva en sí la cabecera con los datos de control.

Ping cada 0.5 segundos

ping 192.168.57.102 –i 0.5

Una vez estudiado el comportamiento de los protocolos con un tamaño de paquete superior, lo siguiente a estudiar era el comportamiento cuando el envío de paquetes era más rápido. Inicialmente, en vez de un paquete cada segundo, se enviarían dos paquetes cada segundo. Esto es, un paquete cada 0.5 segundos. Para ello, se añade al comando ping el parámetro –i 0.5. El parámetro –i se refiere a interval (intervalo). Enviará un paquete cada 0.5 segundos.

Ping cada 0.1 segundos

ping 192.168.57.102 –i 0.1

Por último, se aumenta más aún la frecuencia para comprobar el comportamiento de ambos protocolos con una carga de trabajo mucho mayor. La herramienta ping no permite el envío de paquetes cada menos tiempo, por lo que ésta es la frecuencia mayor a la que se puede trabajar. Para poder ejecutar este comando, antes hay que estar loggeado como superusuario, sino Linux emitirá un mensaje diciendo que no se tienen los permisos necesarios. Con este nuevo parámetros, se enviaran 10 paquetes cada segundo, ésto es, un paquete cada 0.1 segundos.

Peticiones Web

Si una red merece mención especial en informática, ésta es Internet, la red de redes. Millones de usuarios consultan información en internet accediendo a las páginas web.

El objetivo de esta prueba fue calcular el comportamiento de las peticiones de los usuarios a la información que los servidores a través del ofrecen a los usuarios. Para ello, se utilizó un script que realizaba peticiones web al servidor con una frecuencia de un segundo. Con esta prueba se podía calcular el número de peticiones que se pierden durante el intercambio de red del servidor, y observar la diferencia de tiempos en la red local, y en la red pública a través de cada uno de los dos protocolos. Aquí se adjunta el contenido del script.

```
#!/bin/sh

while true; do
        wget 192.168.57.102
        sleep $1
done
```

Descarga HTTP

Otro de los usos más generalizados de internet es la descarga de ficheros y documentos varios, tales como canciones, películas, artículos, ….

Uno de los protocolos más extendidos en internet, sino el que más, es http (Hypertext Transfer Protocol, o protocolo de transferencia de hipertexto). Es el protocolo utilizado en la world wide web (www). Se basa en transacciones entre un cliente y el servidor. El cliente efectúa una petición hacia el servidor para iniciar la transmisión de un recurso del servidor (información a descargar). El protocolo http para la conexión utiliza el puerto 80.

Para la realización de esta prueba, fue necesario instalar en un host un servidor web, para que la otra máquina pudiese descargar los recursos del mismo. Para ello, se utilizó el servidor web xampp (http://www.apachefriends.org/es/xampp.html). Xampp es una distribución Apache que contiene MySQL, PHP y Perl, y que para el objetivo de la prueba era más que suficiente.

Una vez instalado el servidor web xampp, se procederá a la descarga del recurso a través del siguiente comando:

wget http://192.168.0.129/ComfortablyNumb.mp3

En él, primero utilizamos el comando wget, y a continuación, como parámetro, se pone la dirección IP del servidor http, y la dirección donde se encuentra el recurso a descargar.

SCP

Scp (secure copy, o copia segura) es un medio de transferencia segura de archivos, que usa el protocolo ssh (secure Shell). Esta transferencia se realiza entre dos host en remoto o entre un host en remoto y otro host en local.

Durante la transferencia de datos, éstos son cifrados para evitar posibles extracciones de la información por agentes externos a la comunicación. Scp puede solicitar la contraseña al host para establecer una conexión entre el servidor y el host remoto.

Dadas esas características del protocolo, se consideró que era digno del estudio de este proyecto. A partir del siguiente comando:

scp servidor@192.168.0.129:home/desktop/bigFile

El host realiza una conexión con el servidor, y comienza la descarga del fichero bigFile. Para realizar esta conexión, se solicitará la contraseña, para poder autentificarse.

Scp es un protocolo importante en la copia de ficheros entre host, por lo que su estudio es importante a partir de los protocolos de movilidad que se estudian en éste proyecto. De los factores de la comunicación a estudiar nos interesará ver la diferencia de tiempo en la transferencia cuando se encuentra en la red local y cuando el servidor cambia de red; ver qué ocurre con las conexiones entre los clientes y el servidor cuando éste cambia de red, y cuánto tiempo tardan las conexiones en restablecerse después del cambio de red del servidor.

FTP

FTP (file transfer protocol) es el protocolo de transferencia de ficheros por excelencia. Utiliza para ello habitualmente el puerto 20. Se basa en la transferencia de ficheros entre un servidor y un cliente. Para que ésta transferencia se pueda producir, es necesario instalar el servidor de ftp en una de las máquinas. A día de hoy, la mayoría de las empresas tienen un servidor de ficheros, al que los empleados pueden acceder mediante el protocolo ftp.

El cliente puede coger un fichero desde el servidor, o puede depositar un fichero en el mismo. Para poder realizarlo deberá loggearse antes contra el servidor, y una vez que ya esté realizada la conexión podrá empezar la descarga del fichero. FTP está orientado a conseguir la mayor velocidad de transferencia pero no es tan seguro como lo es scp.

Para poder realizar la conexión, el cliente debe ejecutar el siguiente comando:

¡Error! Referencia de hipervínculo no válida.

A partir de ese momento, el cliente está en disposición de loggearse en el servidor. A continuación hay que introducir el nombre de usuario y la contraseña con la que el usuario puede identificarse en el servidor ftp.

Name (192.168.57.102:ibai): javi
Password:
230 Login successful.

Una vez que el usuario ya está loggeado en el servidor, puede empezar la descarga del fichero. Para ello, es suficiente con utilizar el comando get, seguido de los parámetros direcciónFuente, y direcciónDestino.

ftp> get /home/javi/Escritorio/pequeFTP /home/ibai/Desktop/pequeCopia

Al igual que se hizo con el protocolo scp, a partir de éste comando se estudiarán los tiempos de transferencia del fichero en la red local a través de los protocolos a estudiar, el tiempo de reconfiguración cuando se produce el cambio de red del servidor….

HPING

Por último, y para realizar una prueba un poco diferente, se eligió utilizar hping. Hping es una herramienta útil para realizar auditorías y pruebas sobre una red determinada. Además del envío de paquetes, tanto a nivel tcp, como a nivel udp, tiene una aplicación interesante que todavía no se ha utilizado en el proyecto, y es el escaneo de puertos.

Un puerto lógico es una zona de memoria que el ordenador asocia a un puerto físico, o un canal de comunicación; así, se proporciona un espacio de memoria para el almacenamiento temporal de la información que se quiere transmitir. Hoy en día el uso de puertos está totalmente extendido en la informática, pues todas las comunicaciones que se hacen con la red están redirigidas por los puertos. Por ejemplo, http viaja por el puerto 80, ftp por el puerto 20, telnet por el puerto 23…

Hping permitirá comprobar qué puertos están abiertos y cuáles no. Para ello, hping puede enviar paquetes a cada puerto con una determinada frecuencia, y especificar el primer puerto al que enviar y el último, con el fin de comprobar qué puertos están abiertos y cuáles no, dependiendo de la respuesta obtenida. Para ello, se utilizará el siguiente comando:

hping3 -S 192.168.57.102 -p ++1 -V –fast

Esto hace un SYN inicialmente al puerto 1 del pc sobre el que se hace el estudio, y va aumentando el puerto en cada iteración.

Una vez definida todas las pruebas, se procede a la realización de las mismas definidas tanto sobre máquinas virtuales emulando la estructura definida anteriormente, como sobre una estructura real, también ya definida, con máquinas y routers reales, para poder tener una diversidad con los resultados obtenidos que permita que el estudio desarrollado tenga una buena base en la que sustentarse.

PRUEBAS

En esta sección se detallan los resultados de las pruebas que se han realizado para sacar las conclusiones de este proyecto. Las primeras pruebas realizadas han sido tanto para entrar en contacto con el entorno de trabajo como para ir comprobando que los pasos que se daban eran correctos. A partir de este punto, se ejecutarán las pruebas que se han detallado en el punto anterior, y que se utilizarán como base para evaluar tanto VPN como HIP, e intentar decidir cuál se ha comportado mejor desde la experiencia proporcionada por las pruebas mencionadas.

Para que sea más sencillo seguir las pruebas, se adjunta a continuación una tabla con las pruebas realizadas y su orden. También se añade una lista con todos los puntos de los que constan las pruebas, para una mejor identificación.

PING	VPN	Estructura Virtual
		Estructura Real
	HIP	Estructura Virtual
		Estructura Real
PETICIONES WEB	VPN	Estructura Virtual
		Estructura Real
	HIP	Estructura Virtual
		Estructura Real
DESCARGA HTTP	VPN	Estructura Virtual
		Estructura Real
	HIP	Estructura Virtual
		Estructura Real
SCP	VPN	Estructura Virtual
		Estructura Real
	HIP	Estructura Virtual
		Estructura Real
FTP	VPN	Estructura Virtual
		Estructura Real
	HIP	Estructura Virtual
		Estructura Real
HPING	VPN	Estructura Virtual
	HIP	Estructura Real

Tabla 1 : Tabla con las distintas pruebas realizadas

1 – Ping
- 1.1 – VPN
  - 1.1.1 Virtual
  - 1.1.2 Real
- 1.2 – HIP
  - 1.2.1 Virtual
  - 1.2.2 Real

2 – Peticiones Web
- 2.1 – VPN
  - 2.1.1 Virtual
  - 2.1.2 Real
- 2.2 – HIP
  - 2.2.1 Virtual
  - 2.2.2 Real

3 – Descarga HTTP
- 3.1 – VPN
  - 3.1.1 Sin cambiar de red
    - 3.1.1.1 Virtual
    - 3.1.1.2 Real
  - 3.1.2 Cambiando de red
    - 3.1.2.1 Virtual
    - 3.1.2.2 Real
- 3.2 – HIP
  - 3.2.1 Sin cambiar de red
    - 3.2.1.1 Virtual
    - 3.2.1.2 Real
  - 3.2.2 Cambiando de red
    - 3.2.2.1 Virtual
    - 3.2.2.2 Real

4 – SCP
- 4.1 – VPN
  - 4.1.1 Red privada
    - 4.1.1.1 Virtual
    - 4.1.1.2 Real
  - 4.1.2 Red pública
    - 4.1.2.1 Virtual
    - 4.1.2.2 Real
  - 4.1.3 Cambio de red
    - 4.1.3.1 Virtual
    - 4.1.3.2 Real
- 4.2 – HIP
  - 4.2.1 Red privada
    - 4.2.1.1 Virtual
    - 4.2.1.2 Real
  - 4.2.2 Red pública
    - 4.2.2.1 Virtual
    - 4.2.2.2 Real
  - 4.2.3 Cambio de red
    - 4.2.3.1 Virtual
    - 4.2.3.2 Real

5 – FTP
- 5.1 – VPN
  - 5.1.1 Virtual
    - 5.1.1.1 Red Local
    - 5.1.1.2 Red Virtual
  - 5.1.2 Real
    - 5.1.2.1 Red Local
    - 5.1.2.2 Red Virtual
- 5.2 – HIP
  - 5.2.1 Virtual
    - 5.2.1.1 Red Local
    - 5.2.1.2 Red Virtual
  - 5.2.2 Real
    - 5.2.2.1 Red Local
    - 5.2.2.2 Red Virtual

6 – HPING
- 6.1 – VPN
- 6.2 – HIP

1.- Ping

Pruebas realizadas

En este apartado se especifican las pruebas realizadas utilizando hip y openvpn, enviando paquetes de datos mediante ping desde el cliente hasta el servidor. En primer lugar se describe brevemente cómo se han llevado a cabo las pruebas, para después tratar los resultados extraídos de ellas.

En resumen, las pruebas realizadas para esta parte del proyecto, como ya se especificó en el apartado anterior, han sido las siguientes:

- Envío de paquetes ping básicos durante el cambio de red.
- Envío de paquetes ping de 1000 bytes durante el cambio de red.
- Envío de paquetes ping cada 0.5 segundos durante el cambio de red.
- Envío de paquetes ping cada 0.1 segundos durante el cambio de red.

Objetivo de la prueba

Para la realización de esta prueba se tuvieron que utilizar tres máquinas. La primera de ellas era una máquina con dos interfaces de red (en vpn, función de openvpn; en hip, sólo de router); otra era el servidor de datos (en vpn, Cliente VPN), que cambiaría de red, y una última máquina que se encontraría en la red local, y cuya función será cliente del servidor.

Empezaba la transmisión de datos entre el servidor (cliente openvpn) y el cliente, y en un momento dado, se realizaba un cambio de red del servidor a otra red distinta de la que se encuentra el cliente, quedando el servidor en otra red, pero conectada a la primera red a través de una de las tecnologías de movilidad sobre las que se realiza el proyecto, para que la transmisión de paquetes siguiera su curso.

El motivo de hacer diversas pruebas ping sobre las máquinas es ver cómo se comportaba el cambio de red cuando se aplicaba sobre el envío variado de datos, es decir, mandando datos más grandes, o enviando paquetes cada menos tiempo. Para todas las pruebas, se repetía el proceso 5 veces, para tener una mayor fiabilidad en los resultados; y después, para cada uno de ellos, se sacaba la media. De cada prueba se sacaron una serie de datos, útiles para ver el número de paquetes que se perdían en el intercambio, y el tiempo medio de envío de paquetes en una red y en la otra a través de las tecnologías de movilidad.

Para mayor claridad de la prueba, se adjuntan las siguientes ilustraciones, que explican la situación inicial y final de la prueba.

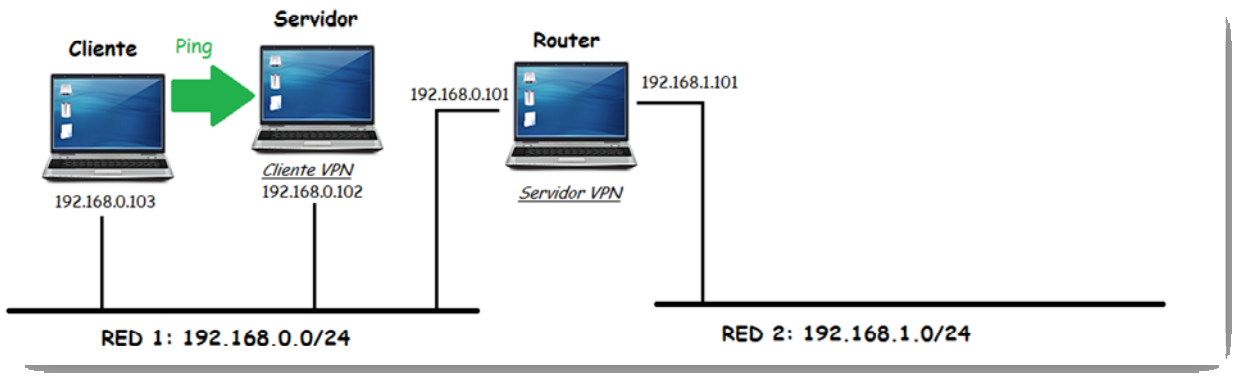


Ilustración 9 : Ping en red local

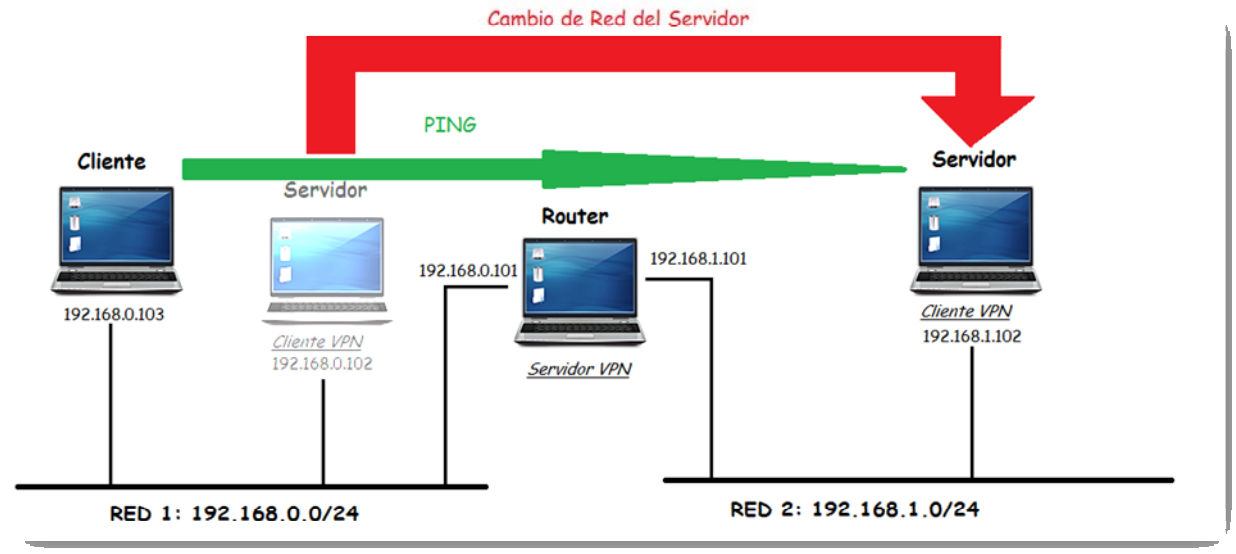


Ilustración 10 : Ping en red pública

1.1 - VPN

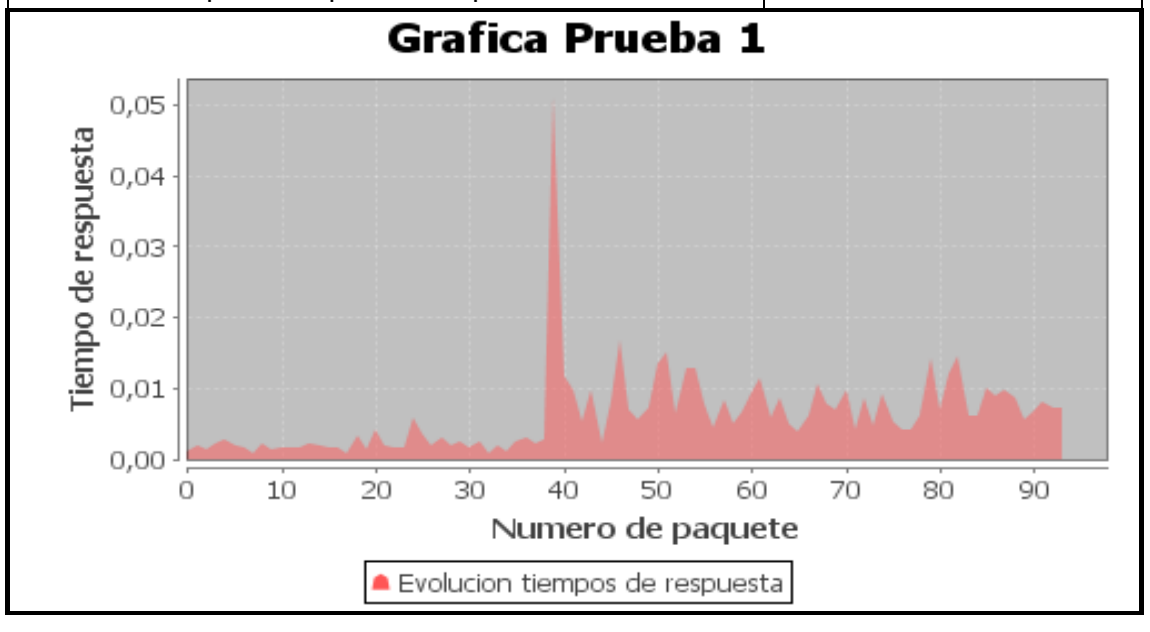
💻1.1.1 - Virtual

1.1.1.1- Envío de paquetes ping básicos durante el cambio de red

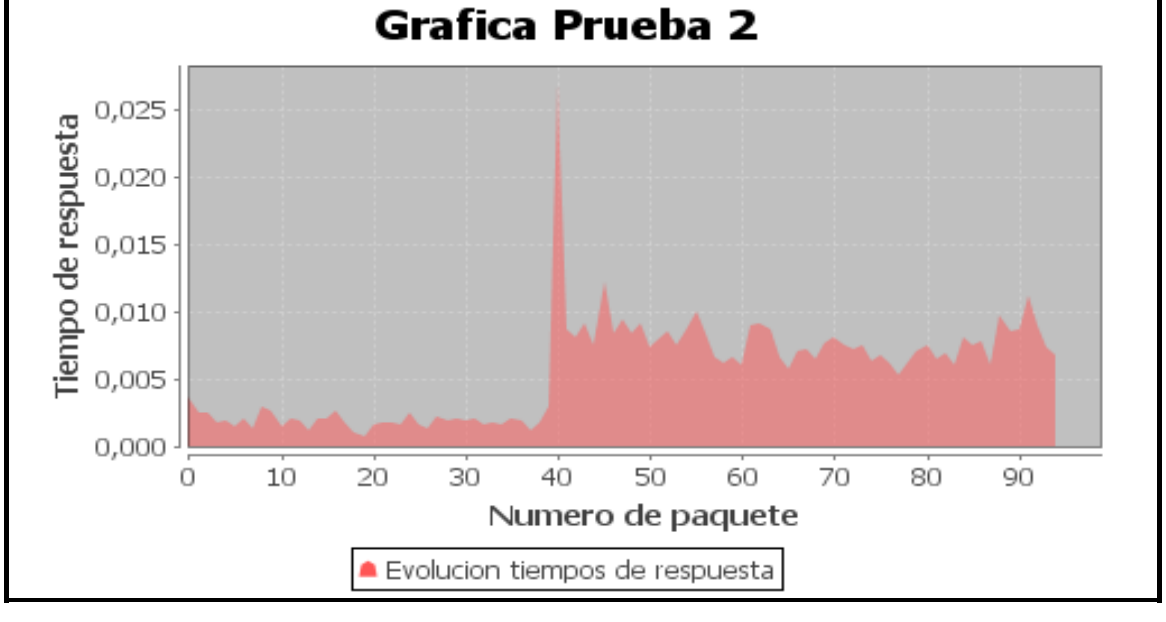
La primera prueba que se realizó fue un ping básico, que consiste en transferir información entre dos pc´s mediante el envío de datos de un tamaño de 64 bytes, y con una frecuencia de envío de un segundo entre paquete y paquete.

Para cada intento de cada prueba se adjuntan los datos obtenidos y una gráfica, para poder ver de forma gráfica el resultado y la diferencia de tiempos entre el envío en red local y a través de la red vpn. Después de los cinco intentos, se adjunta la tabla con la media de éstos, y unas conclusiones.

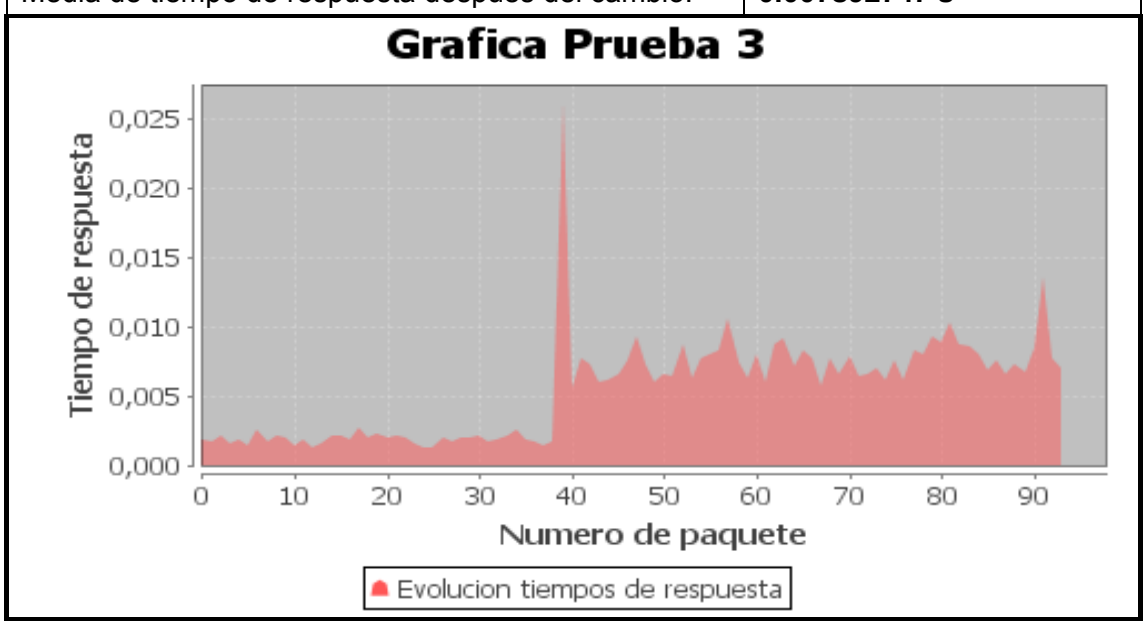
PRUEBA 1.1.1.1.A	
Número total de Request:	**100**
Número total de Reply:	**94**
Número de peticiones perdidas:	**6**
El cambio de red lo efectuó en el paquete:	**39**
Media de tiempo de respuesta antes del cambio:	**0.0020339668 s**
Media de tiempo de respuesta después del cambio:	**0.00885308 s**



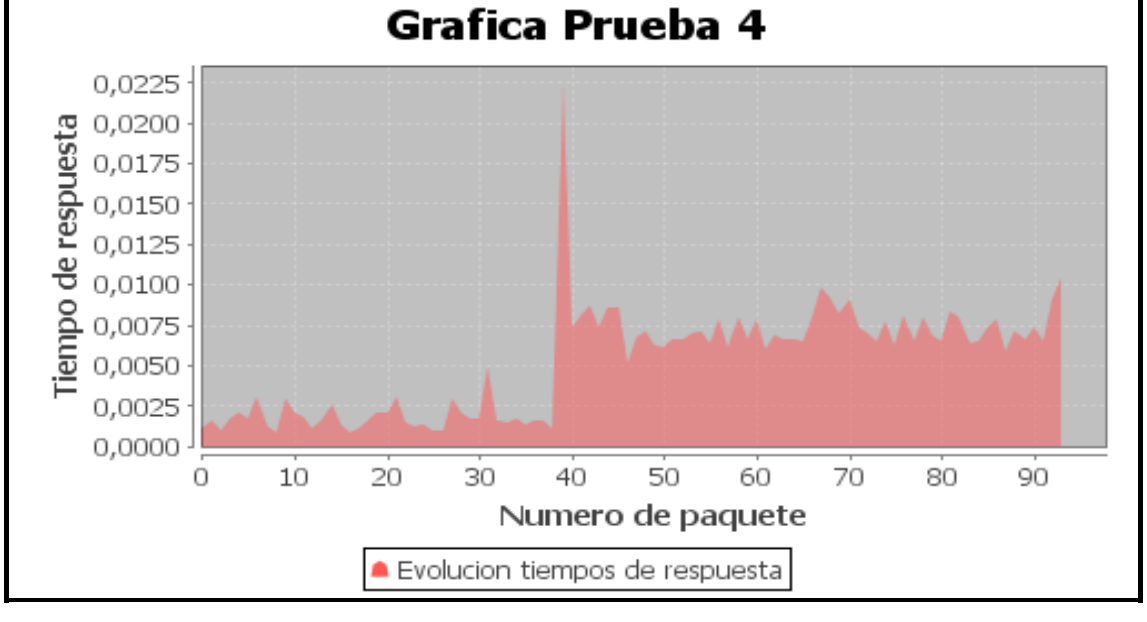
PRUEBA 1.1.1.1.B	
Número total de Request:	**100**
Número total de Reply:	**95**
Número de peticiones perdidas:	**5**
El cambio de red lo efectuó en el paquete:	**40**
Media de tiempo de respuesta antes del cambio:	**0.0018621634 s**
Media de tiempo de respuesta después del cambio:	**0.008038469 s**



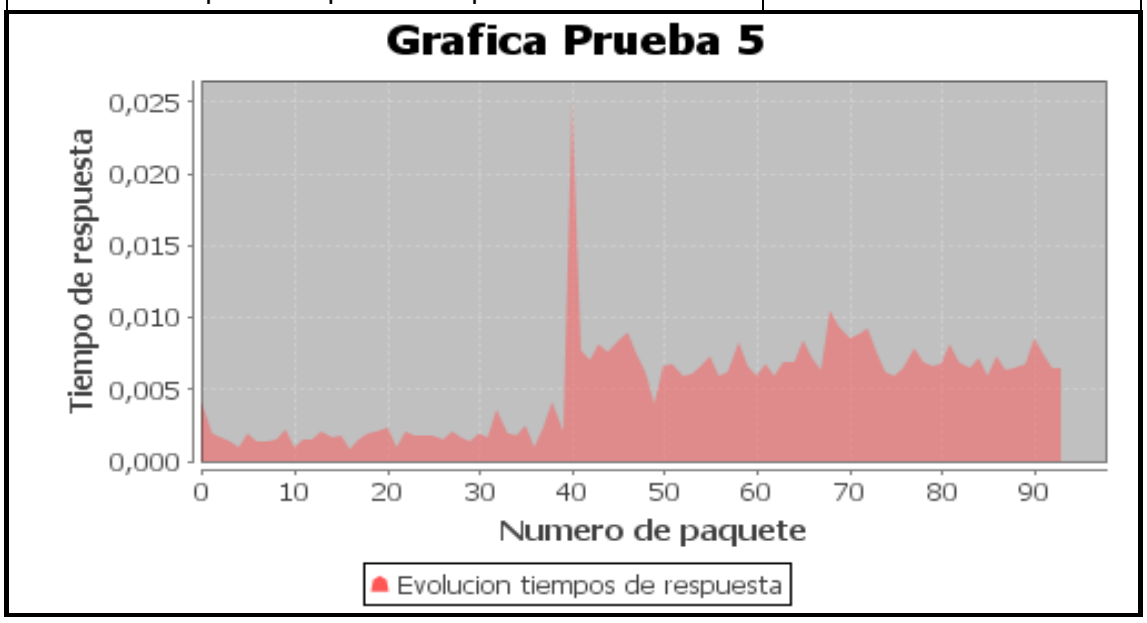
PRUEBA 1.1.1.1.C	
Número total de Request:	**100**
Número total de Reply:	**94**
Número de peticiones perdidas:	**6**
El cambio de red lo efectuó en el paquete:	**39**
Media de tiempo de respuesta antes del cambio:	**0.0018305632 s**
Media de tiempo de respuesta después del cambio:	**0.007892747 s**



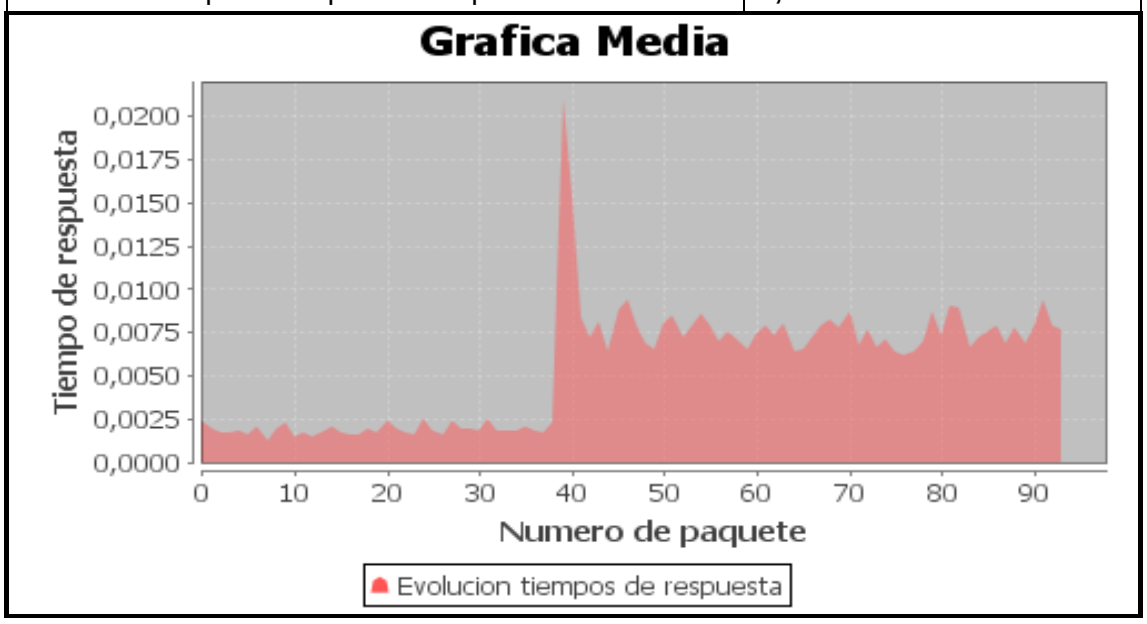
PRUEBA 1.1.1.1.D	
Número total de Request:	**100**
Número total de Reply:	**94**
Número de peticiones perdidas:	**6**
El cambio de red lo efectuó en el paquete:	**39**
Media de tiempo de respuesta antes del cambio:	**0.0017036187 s**
Media de tiempo de respuesta después del cambio:	**0.007547136 s**



PRUEBA 1.1.1.1.E	
Número total de Request:	**100**
Número total de Reply:	**94**
Número de peticiones perdidas:	**6**
El cambio de red lo efectuó en el paquete:	**40**
Media de tiempo de respuesta antes del cambio:	**0.0018111244 s**
Media de tiempo de respuesta después del cambio:	**0.0074040447 s**



MEDIA DE LAS PRUEBA 1.1.1.1: PING NORMAL VPN ESTRUCTURA VIRTUAL	
El Número de peticiones perdidas:	**5.8**
Media de tiempo de respuesta antes del cambio:	**0,00170831162 s**
Media de tiempo de respuesta después del cambio:	**0,00794709534 s**



Conclusiones

Se puede observar que el número de paquetes que se pierde es pequeño, pero hay una pequeña diferencia respecto al tiempo de recepción del paquete entre estar en una red local o estar conectado a través de la red VPN. En este caso existe una diferencia de unas 6 milésimas de media entre el envío y la recepción de cada paquete. Esto es normal, pues a través de la red VPN, además de medidas de seguridad, los paquetes tienen que pasar por más routers y pc´s que en una red local.

También se observa que en el momento del cambio de la red, existe un pico. Esto se debe a que una vez hecho el cambio de red, y por tanto una vez cambiada la dirección IP del server, la máquina local debe hacer una petición ARP para actualizar la información sobre el otro extremo de la comunicación para poder proseguir con la transferencia de datos.

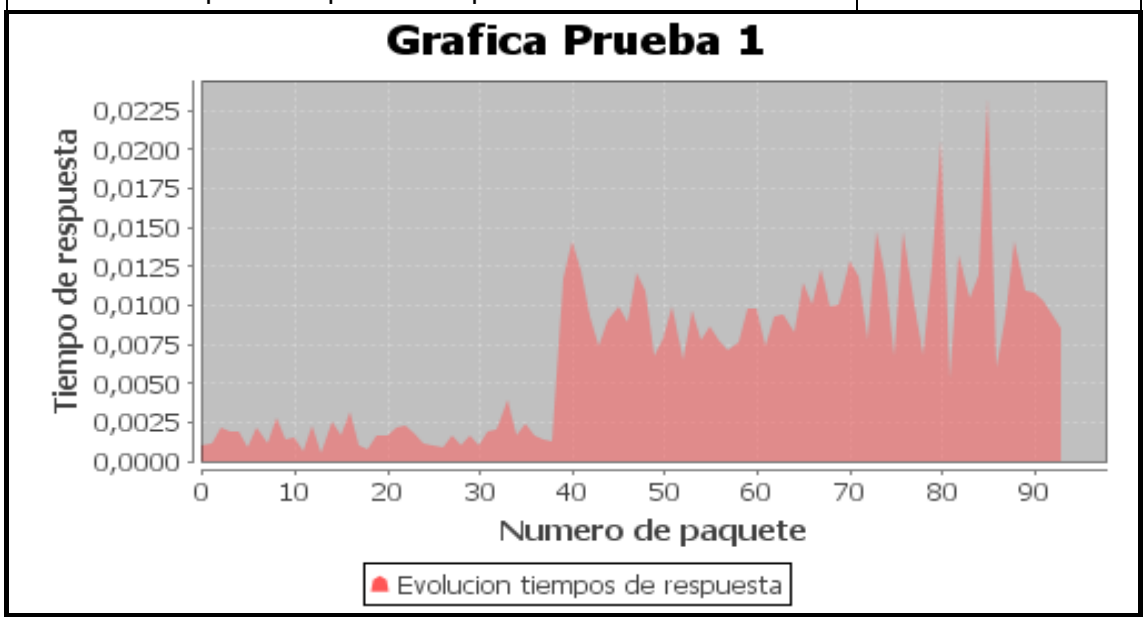
Por último, se concluye que el tiempo que tarda en establecerse la conexión está en torno a los 6 segundos, desde que se recibe el último paquete en la red local hasta que se recibe el primero en la otra red.

1.1.1.2 Envío de paquetes pings de 1000 bytes durante el cambio de red

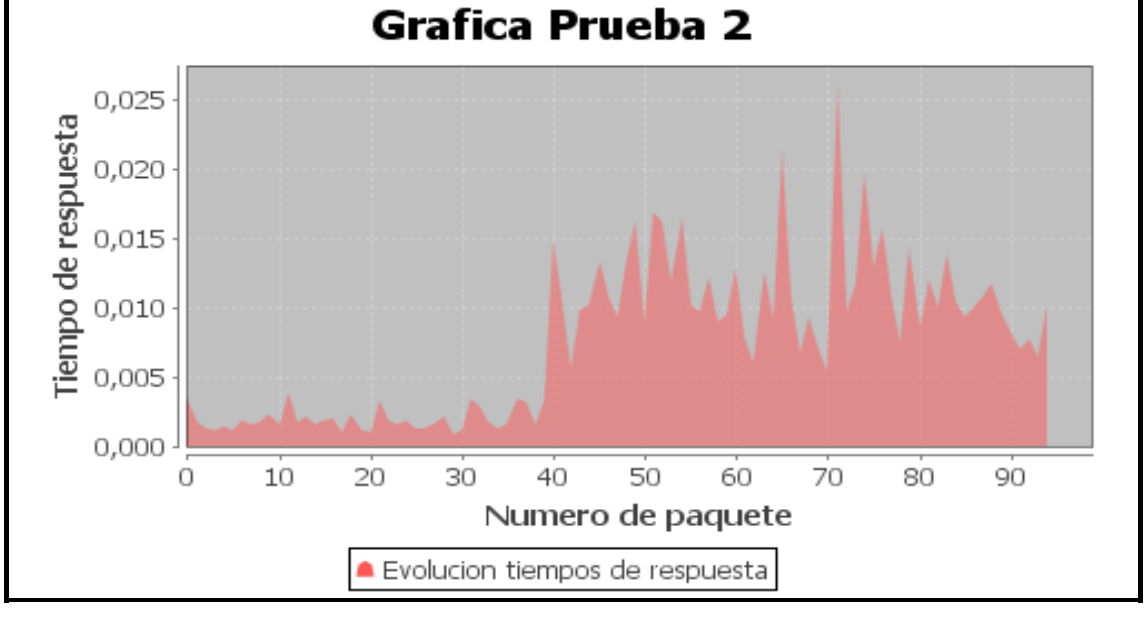
Después de esta primera prueba básica del ping, las pruebas cambiaron para ver el comportamiento de la red con otras características en el envío de paquete. La primera variación consistía en aumentar el tamaño de cada paquete de 64 bytes a 1000 bytes, un tamaño de paquete mucho mayor y con el que se podría estudiar el comportamiento de las tecnologías en una situación mucho más interesante, pues el tamaño de 64 bytes es demasiado pequeño.

Al igual que antes, se extrajeron 5 intentos para mayor exactitud de los resultados, de los cuales se incluyen los datos y las gráficas.

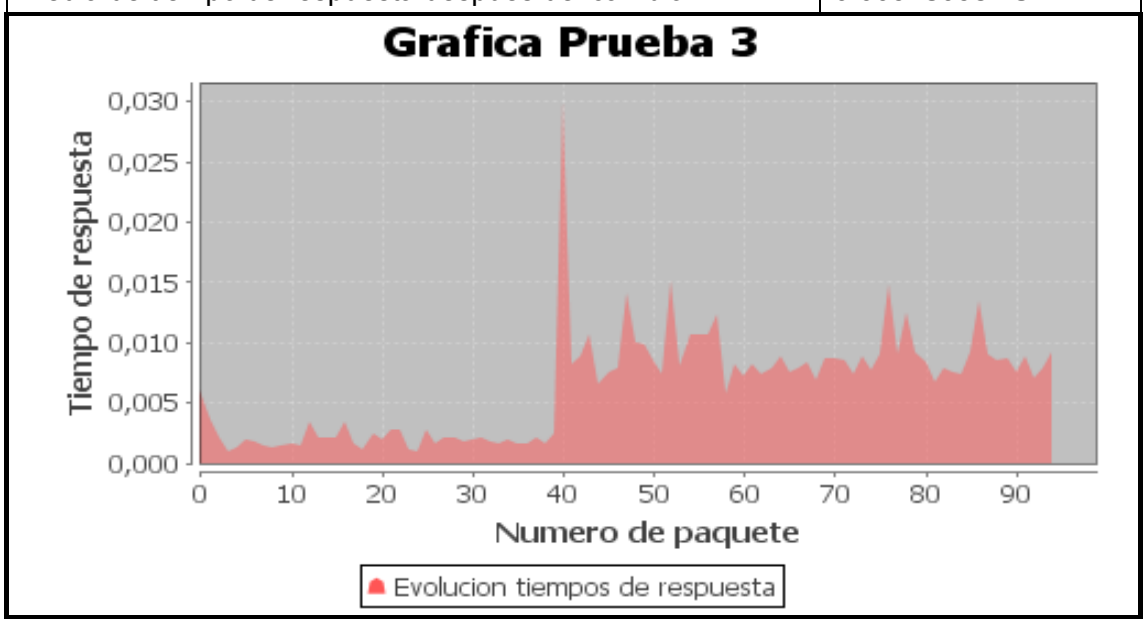
PRUEBA 1.1.1.2.A	
Número total de Request:	**100**
Número total de Reply:	**94**
Número de peticiones perdidas:	**6**
El cambio de red lo efectuó en el paquete:	**39**
Media de tiempo de respuesta antes del cambio:	**0.0016059383 s**
Media de tiempo de respuesta después del cambio:	**0.010265212 s**



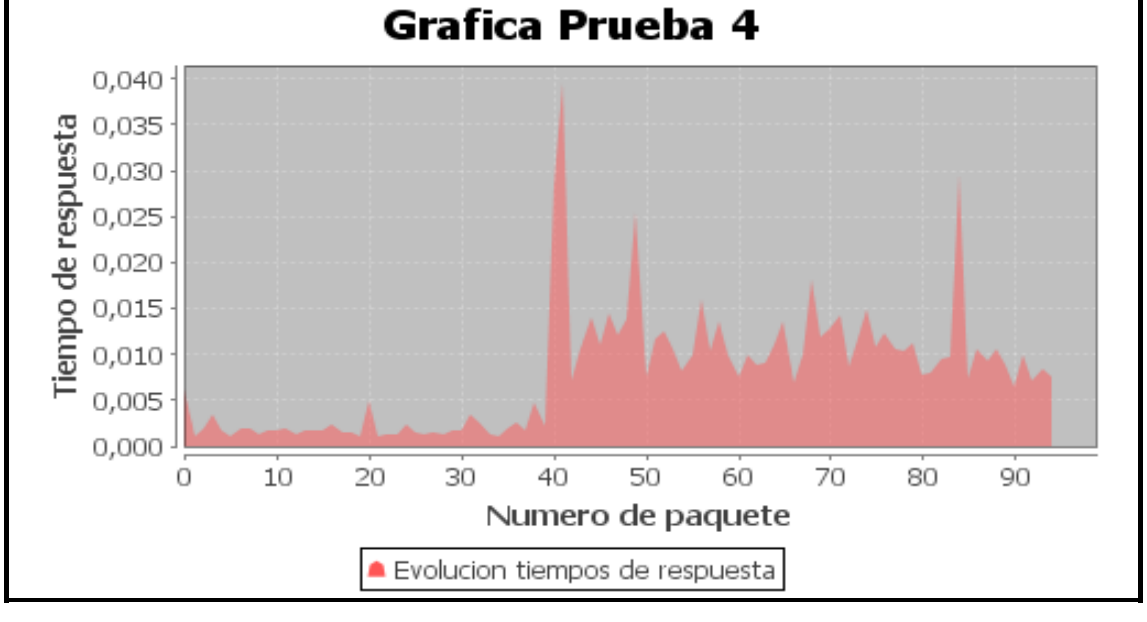
PRUEBA 1.1.1.2.B	
Número total de Request:	**100**
Número total de Reply:	**95**
Número de peticiones perdidas:	**5**
El cambio de red lo efectuó en el paquete:	**40**
Media de tiempo de respuesta antes del cambio:	**0.0018702823 s**
Media de tiempo de respuesta después del cambio:	**0.01119135 s**



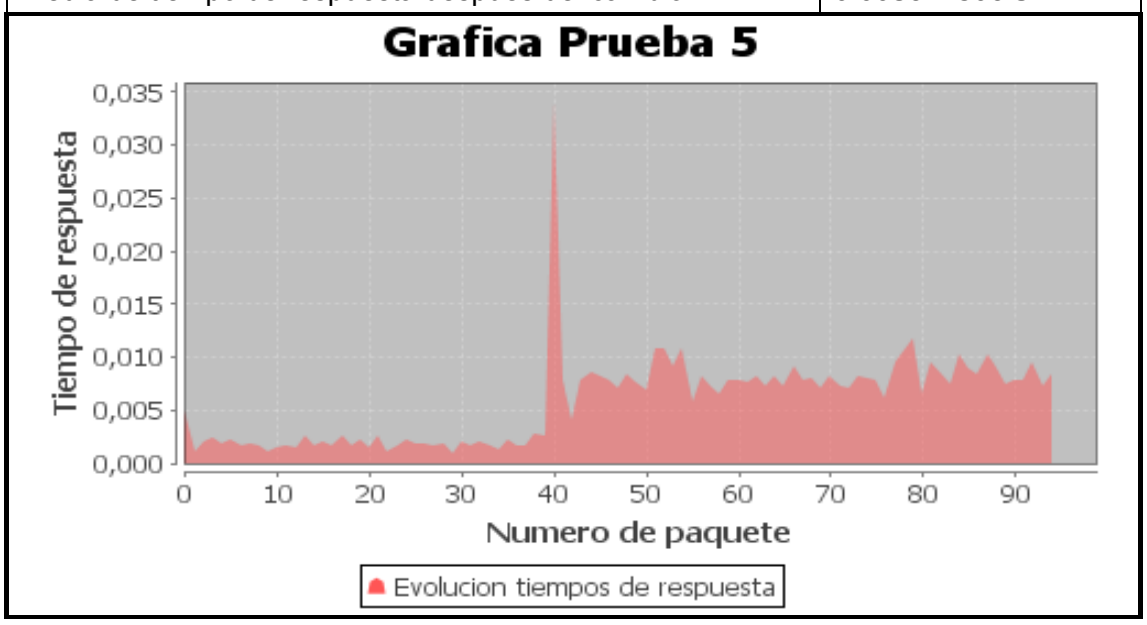
PRUEBA 1.1.1.2.C	
Número total de Request:	**100**
Número total de Reply:	**95**
Número de peticiones perdidas:	**5**
El cambio de red lo efectuó en el paquete:	**40**
Media de tiempo de respuesta antes del cambio:	**0.0020200151 s**
Media de tiempo de respuesta después del cambio:	**0.009280951 s**



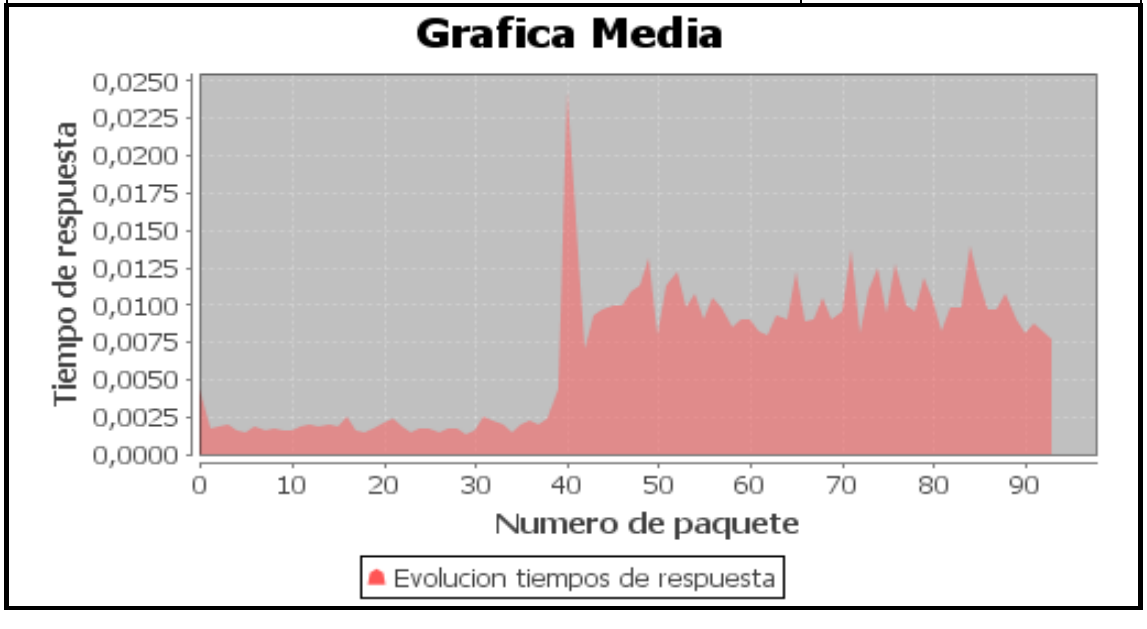
PRUEBA 1.1.1.2.D	
Número total de Request:	**100**
Número total de Reply:	**95**
Número de peticiones perdidas:	**5**
El cambio de red lo efectuó en el paquete:	**40**
Media de tiempo de respuesta antes del cambio:	**0.0019341154 s**
Media de tiempo de respuesta después del cambio:	**0.011957481 s**



PRUEBA 1.1.1.2.E	
Número total de Request:	**100**
Número total de Reply:	**95**
Número de peticiones perdidas:	**5**
El cambio de red lo efectuó en el paquete:	**40**
Media de tiempo de respuesta antes del cambio:	**0.0018580593 s**
Media de tiempo de respuesta después del cambio:	**0.008611506 s**



MEDIA DE LAS PRUEBA 1.1.1.2: PING 1000 bytes VPN ESTRUCTURA VIRTUAL	
El Número de peticiones perdidas:	**5.2**
Media de tiempo de respuesta antes del cambio:	**0,001857679362 s**
Media de tiempo de respuesta después del cambio:	**0,0102613 s**



Conclusiones

Ahora se observa que el número de paquetes que se pierde es ligeramente menor que el que se perdía con la prueba del ping básico, pero no es una medida demasiado representativa, pues la mejora es casi insignificante, por lo que se considera que el tamaño del paquete no es un factor dependiente del número de paquetes perdidos. El tiempo que tarda el cambio de red en producirse y volver a existir conexión sigue siendo de unos 6 segundos.

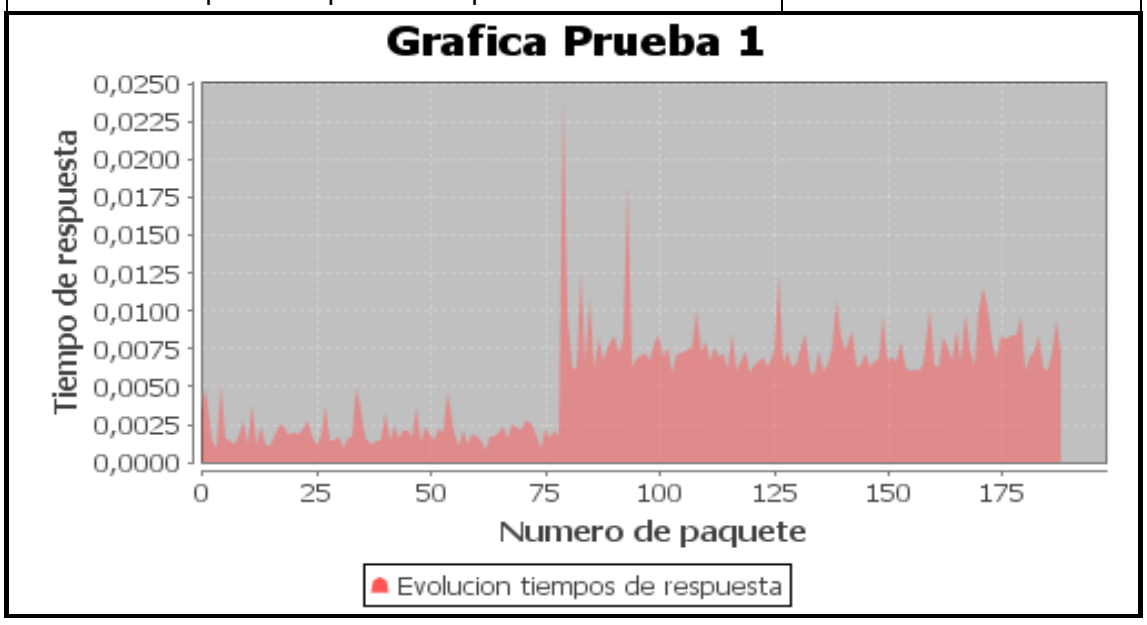
Otro de los detalles que se extraen de la gráfica es que sigue existiendo el pico del tiempo en el cambio de red. Como ya se explicó en la primera prueba, esto es debido a que la máquina local debe hacer una petición ARP, para actualizar la información sobre el otro extremo de la comunicación para poder proseguir con la transferencia de datos. Esto trae consigo el envió de datos a través de la red, y el consecuente retardo en el tiempo.

La variación más importante respecto a la prueba básica reside en el tiempo medio de recepción del paquete, sobre todo en la red que transmite a través de VPN. Si se mira el tiempo medio en la recepción de paquetes en la red local, se aprecia que ha aumentado algo menos de 2 diezmilésimas (un 8.74 % más), lo cual es normal, dado el aumento del tamaño de los paquetes; en cambio, en la segunda red, la de VPN, el aumento es mayor, siendo de más de 2 milésimas, lo que supone un aumento del 29.15%, lo cual sí que supone un aumento considerable.

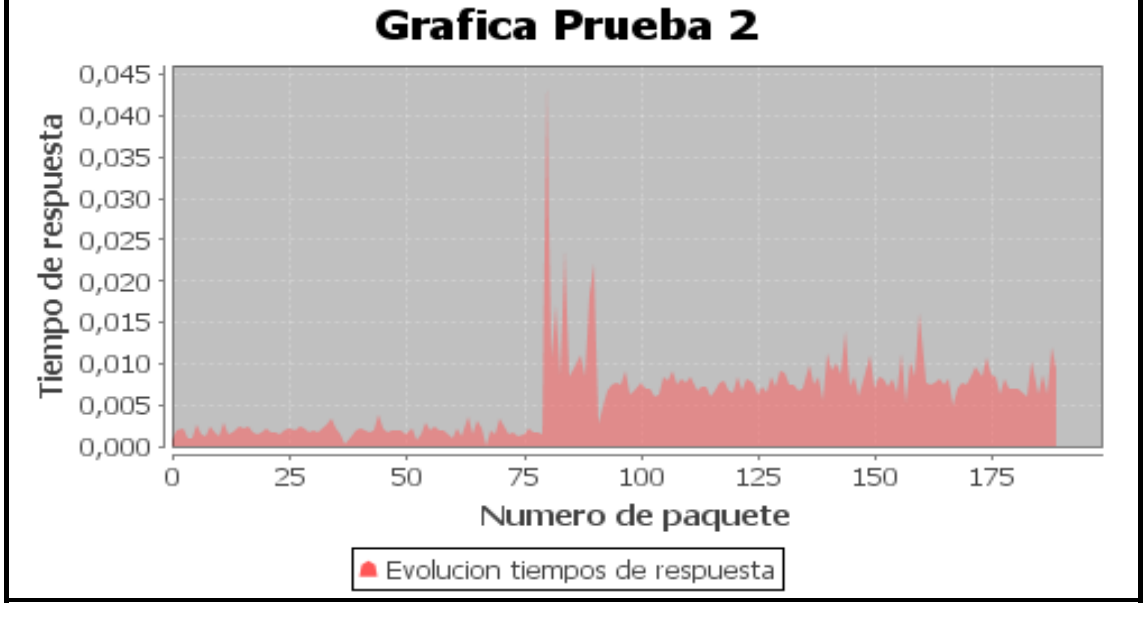
1.1.1.3 Envío de paquetes ping cada 0.5 segundos durante el cambio de red.

Después de probar que ocurría al aumentar el tamaño de los paquetes, se hicieron pruebas disminuyendo el tiempo con el que se enviaban los paquetes, y para ello inicialmente se investigó con una primera prueba en la que el envío se realizaba cada 0.5 segundos. Ahora, al recibir paquetes cada menos tiempo, se afinan más ciertos parámetros, como el tiempo que tarda en restablecerse la recepción de paquetes durante el cambio de red, pues la escala ya no será en segundos, sino en medios segundos. Para que las pruebas tuvieran más relevancia, ahora se duplicó el número de paquetes, de 100 a 200, para tener una mayor muestra, ya que el envío también era más rápido.

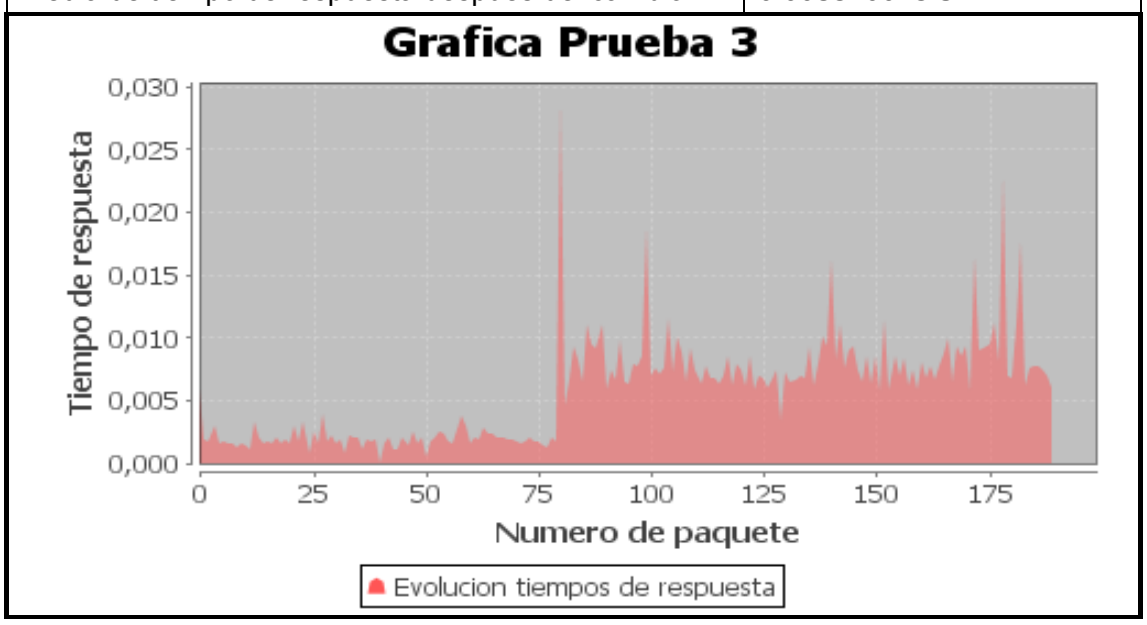
PRUEBA 1.1.1.3.A	
Número total de Request:	**200**
Número total de Reply:	**189**
Número de peticiones perdidas:	**11**
El cambio de red lo efectuó en el paquete:	**79**
Media de tiempo de respuesta antes del cambio:	**0.001988094 s**
Media de tiempo de respuesta después del cambio:	**0.0076964987 s**



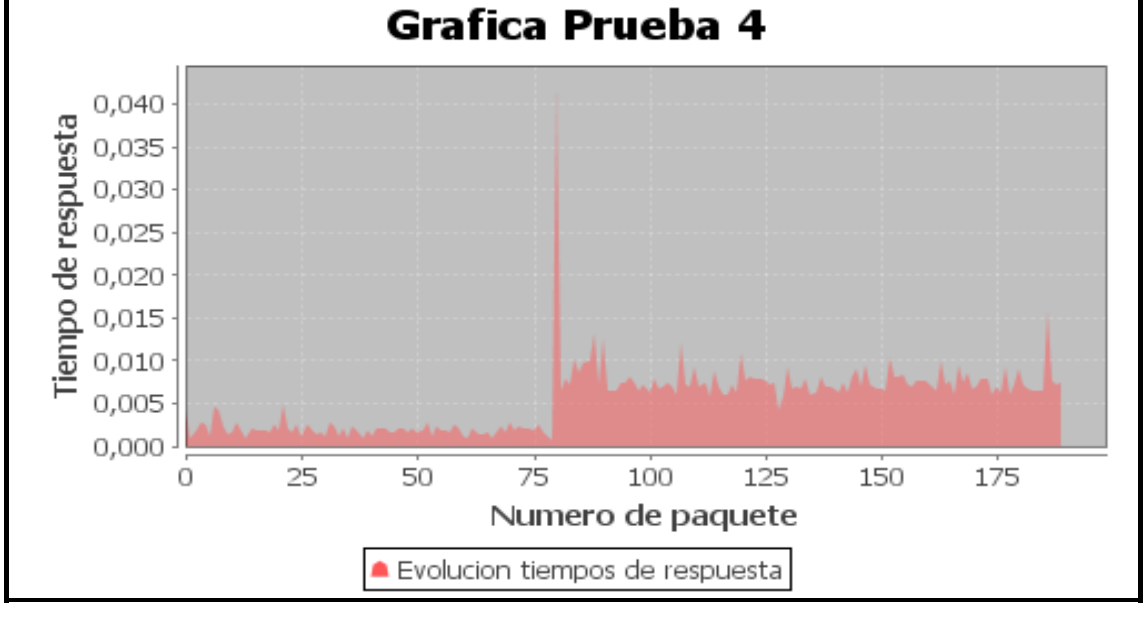
PRUEBA PRUEBA 1.1.1.3.B	
Número total de Request:	**200**
Número total de Reply:	**190**
Número de peticiones perdidas:	**10**
El cambio de red lo efectuó en el paquete:	**80**
Media de tiempo de respuesta antes del cambio:	**0.001771307 s**
Media de tiempo de respuesta después del cambio:	**0.008601241 s**



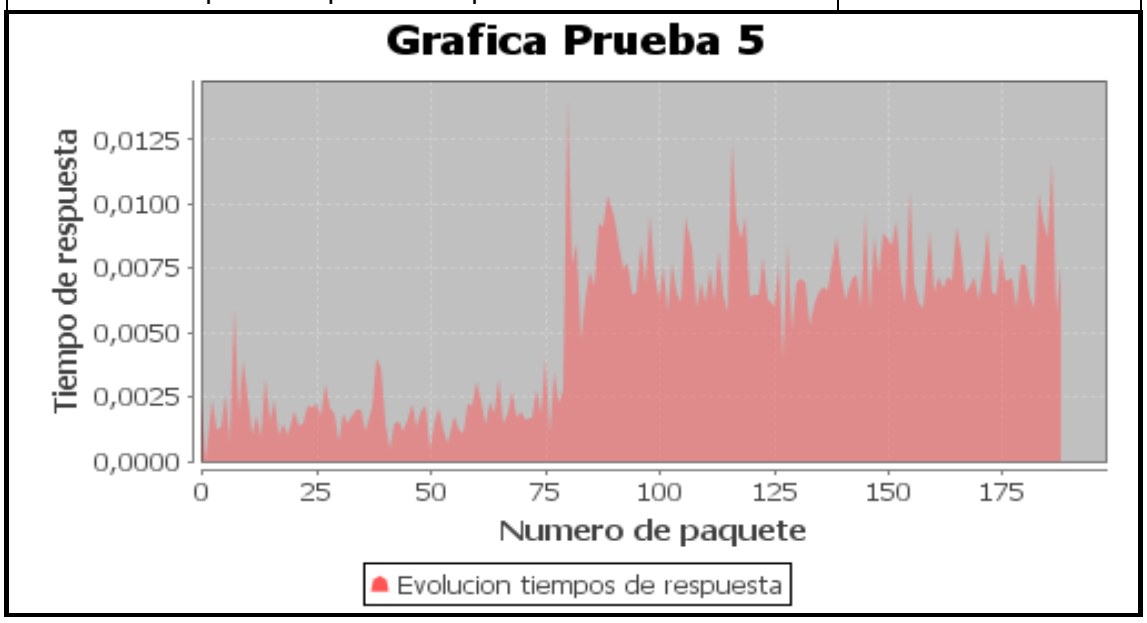
PRUEBA PRUEBA 1.1.1.3.C	
Número total de Request:	**200**
Número total de Reply:	**190**
Número de peticiones perdidas:	**10**
El cambio de red lo efectuó en el paquete:	**80**
Media de tiempo de respuesta antes del cambio:	**0.0018842692 s**
Media de tiempo de respuesta después del cambio:	**0.008340073 s**



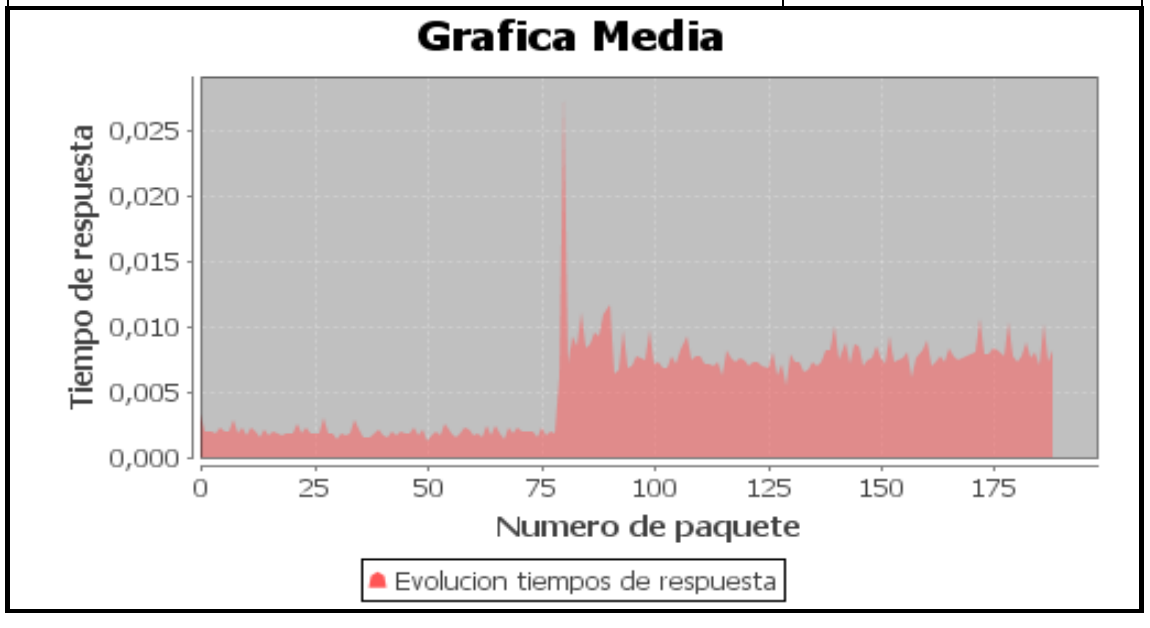
PRUEBA 1.1.1.3.D	
Número total de Request:	**200**
Número total de Reply:	**190**
Número de peticiones perdidas:	**10**
El cambio de red lo efectuó en el paquete:	**80**
Media de tiempo de respuesta antes del cambio:	**0.0018565853 s**
Media de tiempo de respuesta después del cambio:	**0.0078231115 s**



PRUEBA 1.1.1.3.E	
Número total de Request:	**200**
Número total de Reply:	**189**
Número de peticiones perdidas:	**11**
El cambio de red lo efectuó en el paquete:	**80**
Media de tiempo de respuesta antes del cambio:	**0.0019078199 s**
Media de tiempo de respuesta después del cambio:	**0.007447269 s**



MEDIA DE LAS PRUEBA 1.1.1.3: PING 0.5 seg VPN ESTRUCTURA VIRTUAL	
El Número de peticiones perdidas:	**10.4**
Media de tiempo de respuesta antes del cambio:	**0,00188001508 s**
Media de tiempo de respuesta después del cambio:	**0,00798163864 s**



Conclusiones

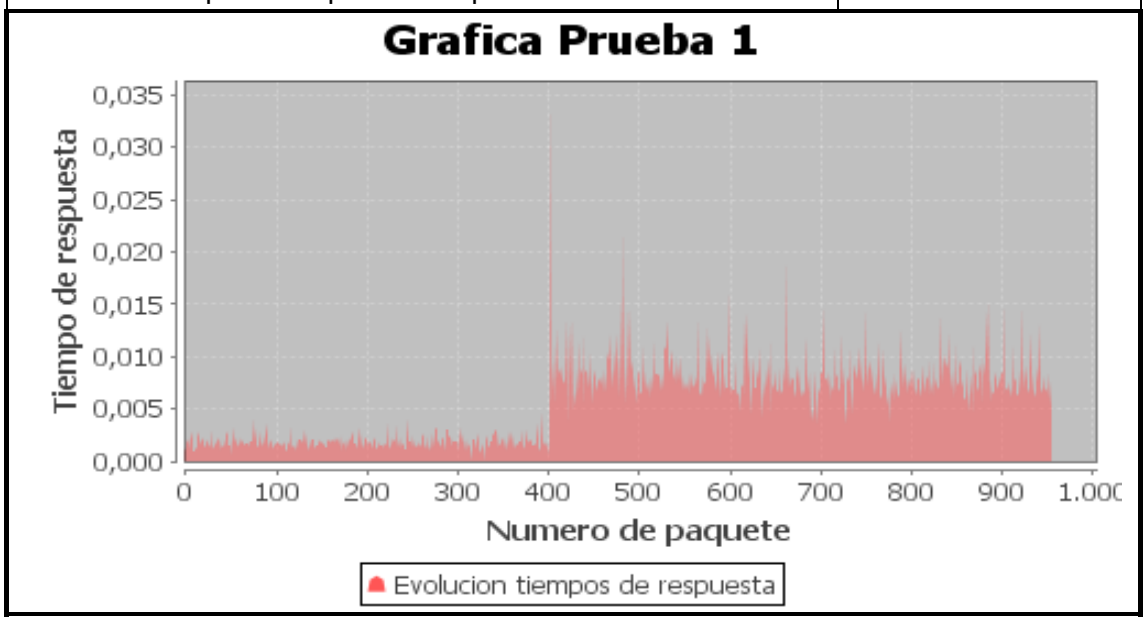
Dado que hemos aumentado la velocidad de envío de los paquetes, es obvio que el número de paquetes perdidos habrá aumentado, pues el tiempo que tarda en cambiar de red y en activar vpn es muy similar, pero no así el envío de paquetes, que sube al doble. Es por eso que se obtiene un número de paquetes perdidos ahora de 10.4. Pero este dato nos está dando que el tiempo que tarda en conectarse está cerca de 5.2 segundos (ya que se envía un paquete cada medio segundo), que afina más que la anterior medida, que nos decía que eran 6 segundos.

En cuanto a la variación del tiempo, tanto en la red local como en la otra red a través de vpn hemos conseguido resultados prácticamente similares (no llegando a las 2 diezmilésimas en ambas redes), lo cual quiere decir que el tiempo es el mismo, y no se ve afectado en cuanto al aumento de la frecuencia en el envío de los datos. El pico sigue estando en el momento del cambio de la red, como sucedía anteriormente, y sube, pero es algo muy puntual, y por la razón ya explicada anteriormente.

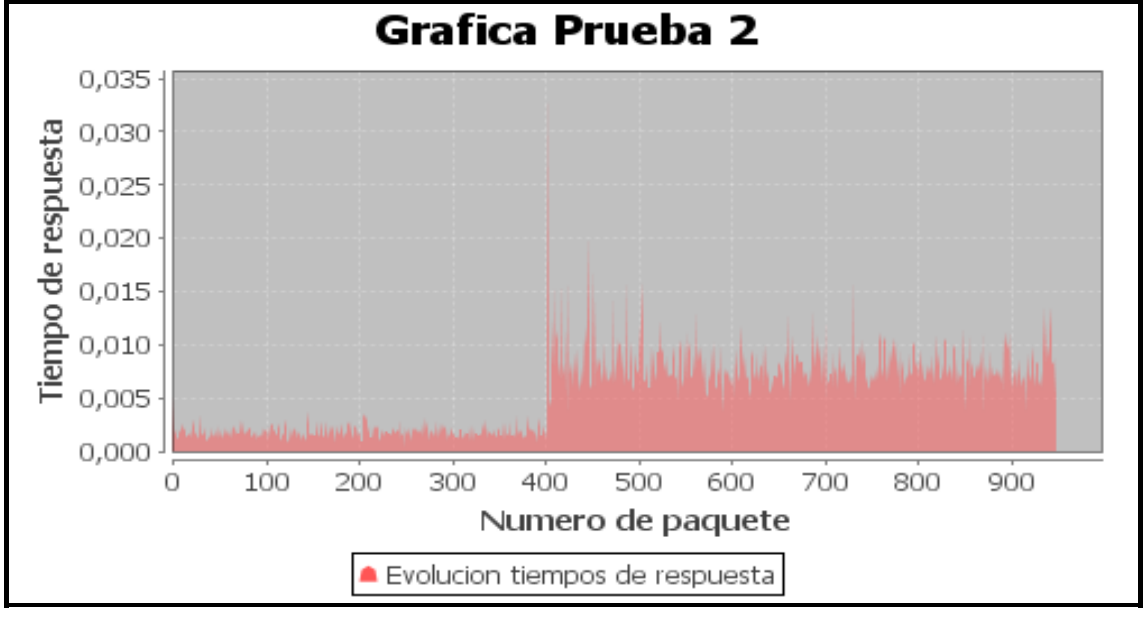
1.1.1.4 Envío de paquetes pings cada 0.1 segundos durante el cambio de red

El objetivo de esta prueba era ver el rendimiento de la red VPN recibiendo una gran cantidad de datos en un breve espacio de tiempo. Para ello, se aumentó la frecuencia de envió, y el tiempo tras el que se realizaba un nuevo envío bajó hasta 0.1 segundos. Para esta frecuencia se tuvieron que cambiar los permisos de usuario de Linux para utilizar permisos de superusuario, pues sino Linux no permitía el envío de datos tan rápido.

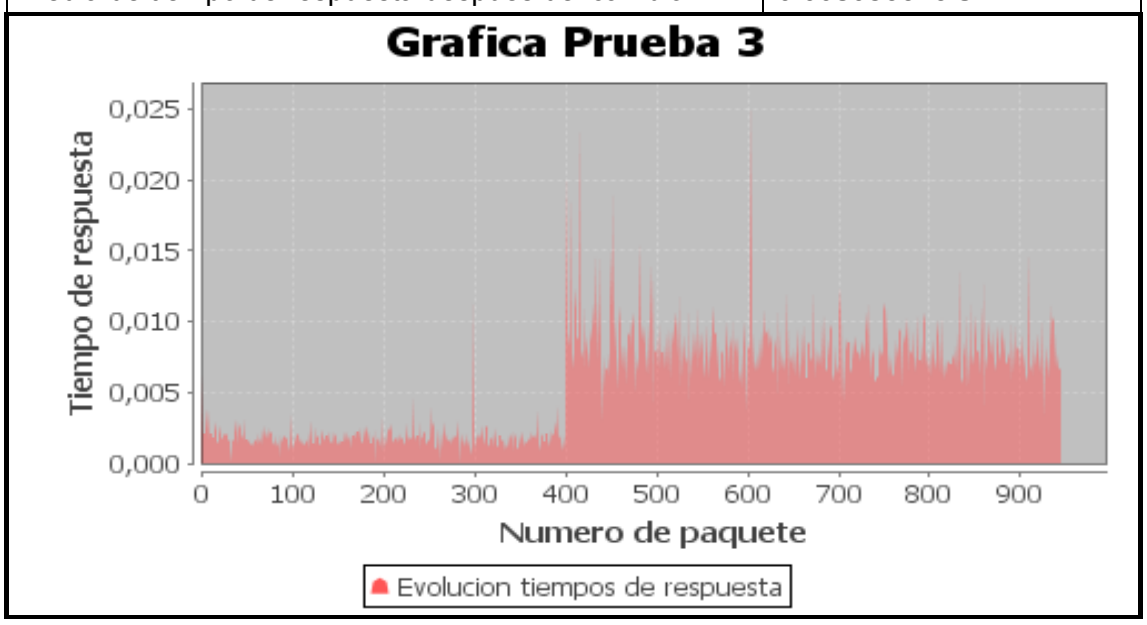
PRUEBA 1.1.1.4.A	
Número total de Request:	**1006**
Número total de Reply:	**956**
Número de peticiones perdidas:	**50**
El cambio de red lo efectuó en el paquete:	**404**
Media de tiempo de respuesta antes del cambio:	**0.0017617308 s**
Media de tiempo de respuesta después del cambio:	**0.008022357 s**



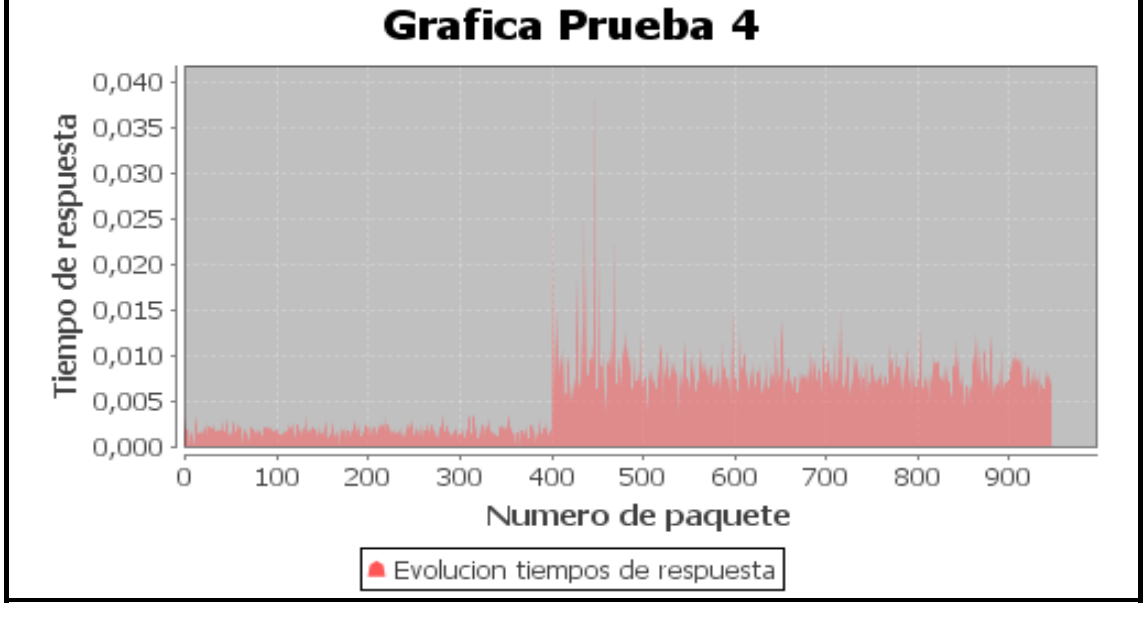
PRUEBA 1.1.1.4.B	
Número total de Request:	**1000**
Número total de Reply:	**949**
Número de peticiones perdidas:	**51**
El cambio de red lo efectuó en el paquete:	**402**
Media de tiempo de respuesta antes del cambio:	**0.0017281296 s**
Media de tiempo de respuesta después del cambio:	**0.00805883 s**



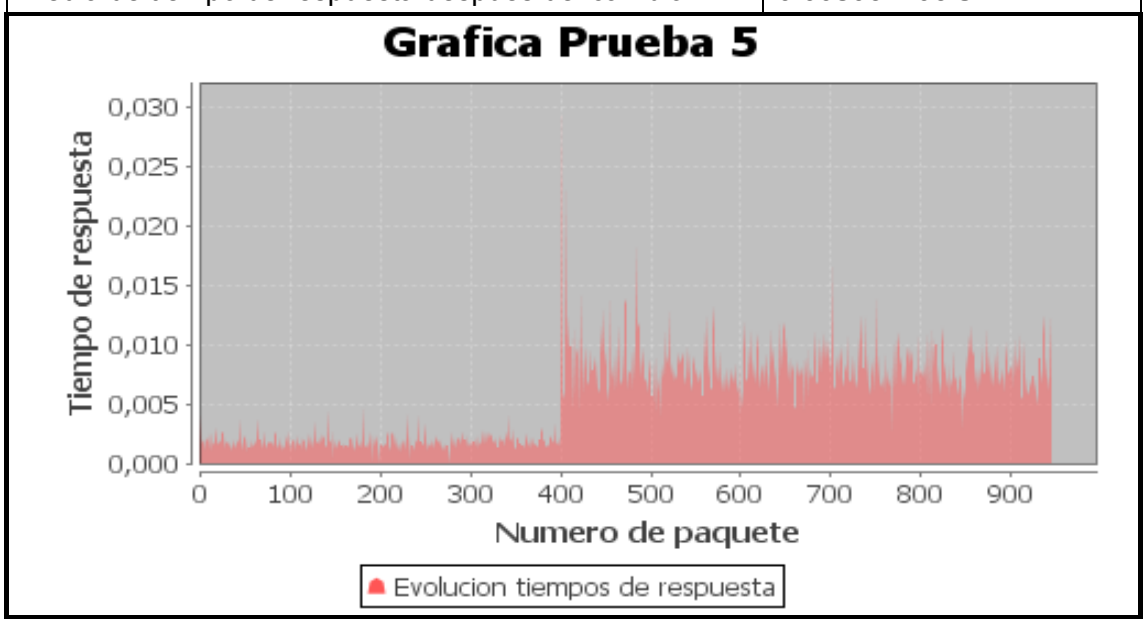
PRUEBA 1.1.1.4.C	
Número total de Request:	**998**
Número total de Reply:	**947**
Número de peticiones perdidas:	**51**
El cambio de red lo efectuó en el paquete:	**401**
Media de tiempo de respuesta antes del cambio:	**0.0017698538 s**
Media de tiempo de respuesta después del cambio:	**0.008056676 s**



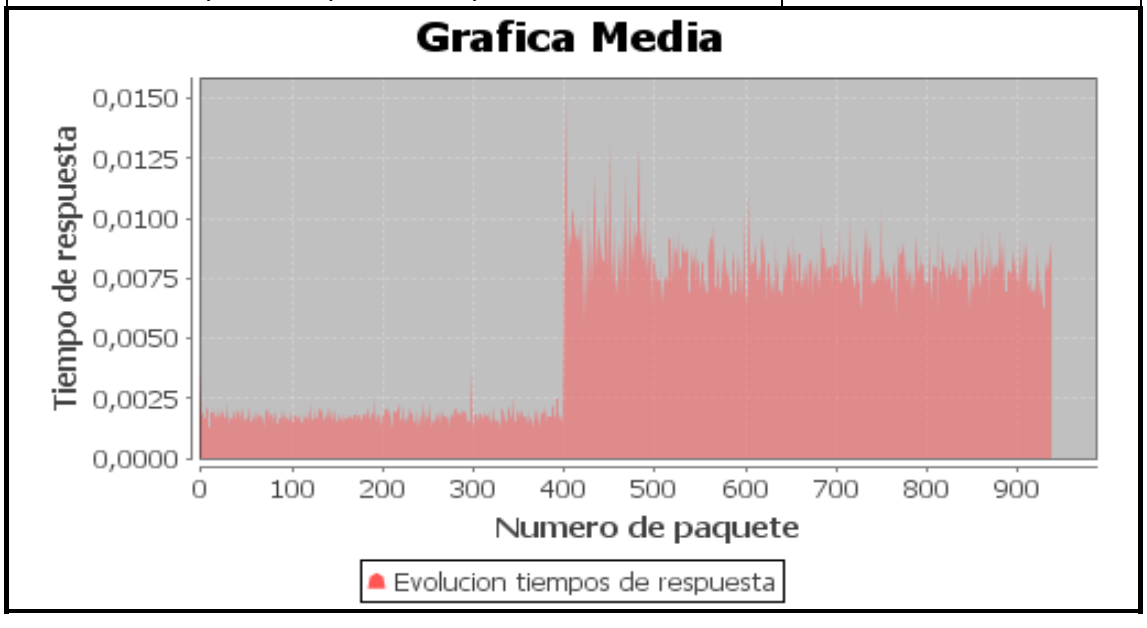
PRUEBA 1.1.1.4.D	
Número total de Request:	**999**
Número total de Reply:	**948**
Número de peticiones perdidas:	**51**
El cambio de red lo efectuó en el paquete:	**403**
Media de tiempo de respuesta antes del cambio:	**0.0016652815 s**
Media de tiempo de respuesta después del cambio:	**0.008142734 s**



PRUEBA 1.1.1.4.E	
Número total de Request:	**999**
Número total de Reply:	**948**
Número de peticiones perdidas:	**51**
El cambio de red lo efectuó en el paquete:	**401**
Media de tiempo de respuesta antes del cambio:	**0.0017429513 s**
Media de tiempo de respuesta después del cambio:	**0.008067206 s**



MEDIA DE LAS PRUEBA 1.1.1.4: PING 0.1 seg VPN ESTRUCTURA VIRTUAL	
El Número de peticiones perdidas::	**50.8**
Media de tiempo de respuesta antes del cambio:	**0,0017335894 s**
Media de tiempo de respuesta después del cambio:	**0,0080695606 s**



Conclusiones

Con esta frecuencia de envío de paquetes, el número de paquetes perdidos ha sido de 50.8 paquetes de media, y dado que cada paquete se enviaba cada 0.1 segundos, se puede establecer que el tiempo medio que tarda en cambiar de red el servidor (cliente vpn) ha sido de algo menos de 5.1 segundos. Esta afinación ha sido posible gracias al aumento de frecuencia en el envió, pues cuando ésta era de 1 paquete por segundo, el tiempo que transcurría desde la desconexión de la primera red a la conexión de la segunda red se medía en segundos, mientras que ahora se mide en décimas de segundo, lo que nos proporciona una mayor afinación.

Respecto al tiempo medio de recepción, tanto antes como después, se obtiene que la diferencia de tiempos es ínfima respecto a la primera prueba, con los valores del ping por defecto (envío de paquetes cada segundo). El tiempo medio en la red local asciende 3 cienmilésimas de segundo (un 1.3 %), y en la red externa a través de vpn asciendo 12 cienmilésimas de segundo, que supone únicamente un 1.5%, por lo que se deduce que openvpn tiene un buen comportamiento con el aumento de frecuencia en el envío de paquetes, no así tanto con el aumento del tamaño de los mismos, como ya se observo en la prueba 2.

1.1.2- Real

Después de realizar todas las pruebas sobre máquinas virtuales, se plasmaron las mismas pruebas sobre una estructura real, con dos routers para tener dos redes independientes, un ordenador con dos tarjetas de red que hacía de router Linux (y servidor vpn) y dos pc's que hacían de host, uno de ellos con la función de servidor (cliente openvpn) que cambiaría de red, y el otro un host que sería un pc fijo en una red, que sería el cliente del servidor anterior. Esta estructura está más detallada en la sección de infraestructuras creadas para las pruebas.

Al igual que con las máquinas virtuales, se desarrollaron cuatro pruebas diferentes con la utilidad diagnóstica Ping, que fueron:

- Envío de paquetes ping básicos durante el cambio de red.
- Envío de paquetes ping de 1000 bytes durante el cambio de red.
- Envío de paquetes ping cada 0.5 segundos durante el cambio de red.
- Envío de paquetes ping cada 0.1 segundos durante el cambio de red.

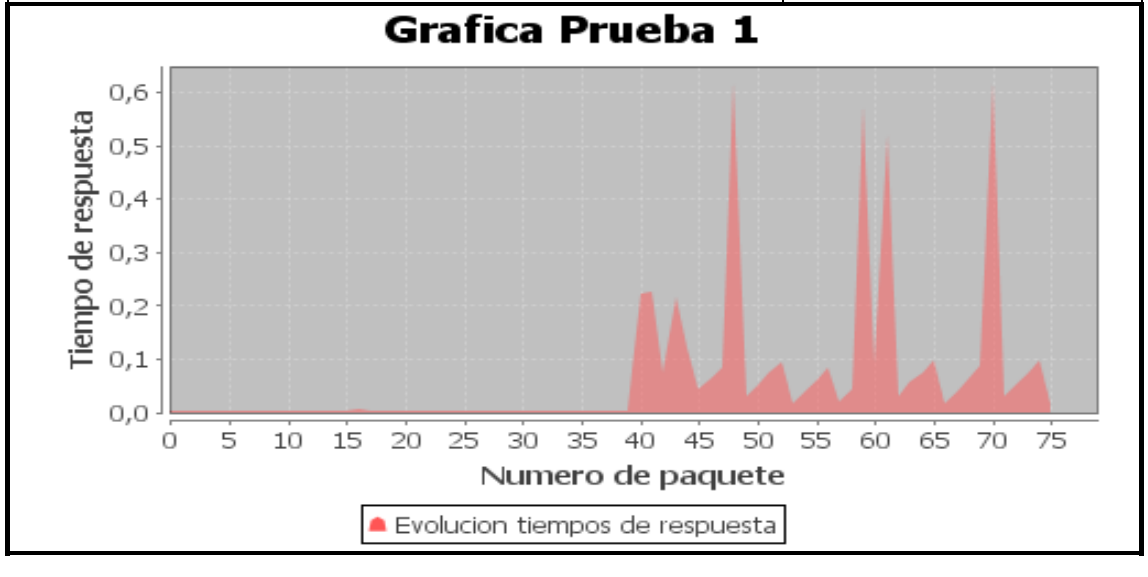
A continuación se detallan los resultados obtenidos para cada una de estas pruebas, con un total de 5 repeticiones para cada una de ellas, y la estadística media de estos 5 intentos.

1.1.2.1 Envío de paquetes ping básicos durante el cambio de red

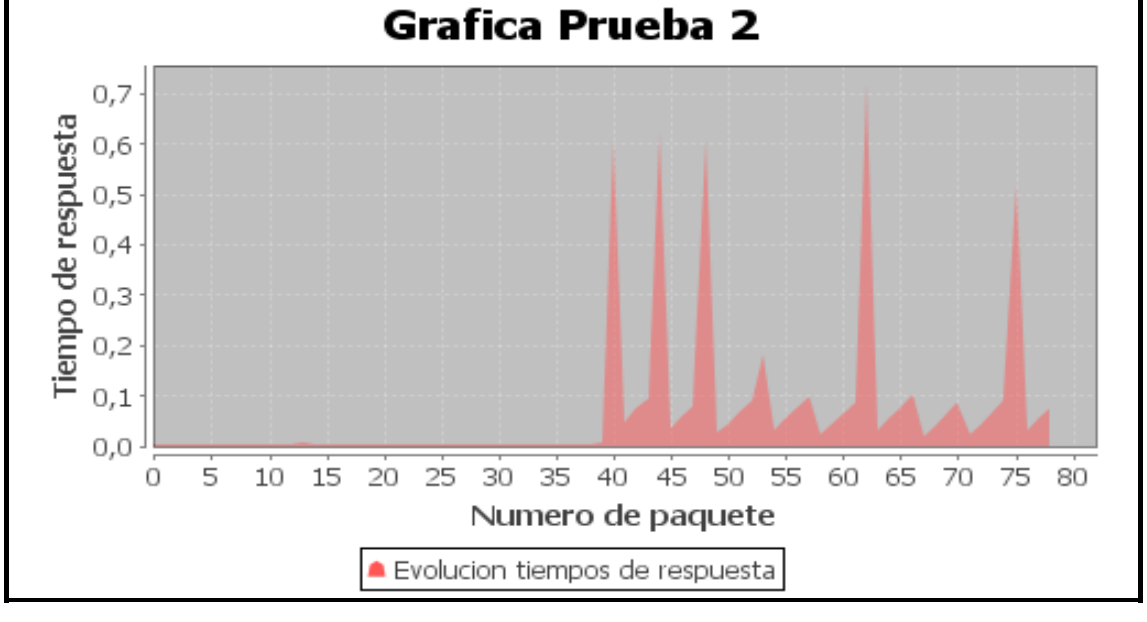
La opción por defecto de esta herramienta de diagnóstico envía paquetes de 64 bytes con una frecuencia de 1 segundo entre paquete y paquete. Es una buena primera medida para observar el comportamiento de openvpn en una estructura real.

Al igual que en el ejemplo anterior, se adjuntan los 5 intentos, con su resumen y su gráfica, y una media de estos 5.

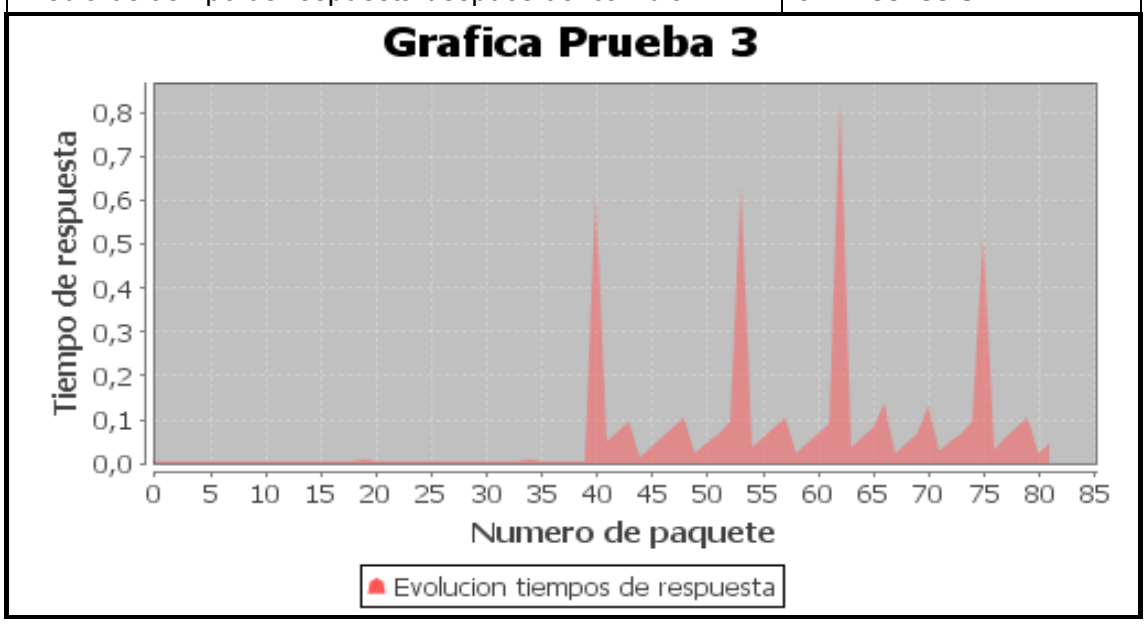
PRUEBA 1.1.2.1.A	
Número total de Request:	**120**
Número total de Reply:	**76**
Número de peticiones perdidas:	**44**
El cambio de red lo efectuó en el paquete:	**40**
Media de tiempo de respuesta antes del cambio:	**0.002372979 s**
Media de tiempo de respuesta después del cambio:	**0.13009793 s**



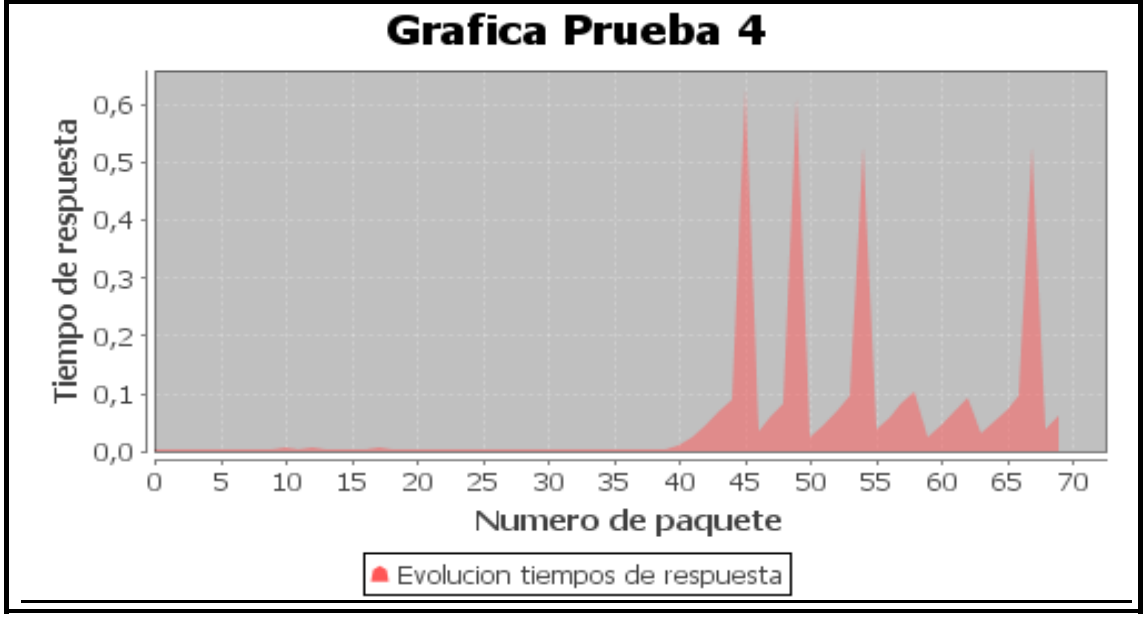
PRUEBA 1.1.2.1.B	
Número total de Request:	**120**
Número total de Reply:	**79**
Número de peticiones perdidas:	**41**
El cambio de red lo efectuó en el paquete:	**40**
Media de tiempo de respuesta antes del cambio:	**0.0024405303 s**
Media de tiempo de respuesta después del cambio:	**0.13263623 s**



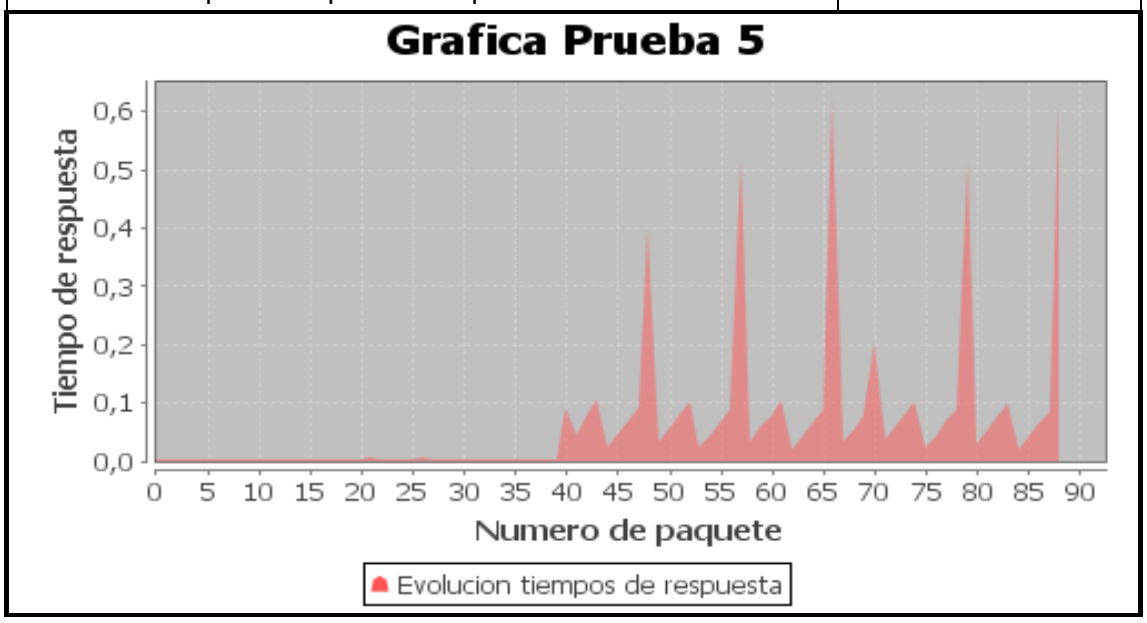
PRUEBA 1.1.2.1.C	
Número total de Request:	**120**
Número total de Reply:	**82**
Número de peticiones perdidas:	**38**
El cambio de red lo efectuó en el paquete:	**40**
Media de tiempo de respuesta antes del cambio:	**0.002614354 s**
Media de tiempo de respuesta después del cambio:	**0.11735135 s**



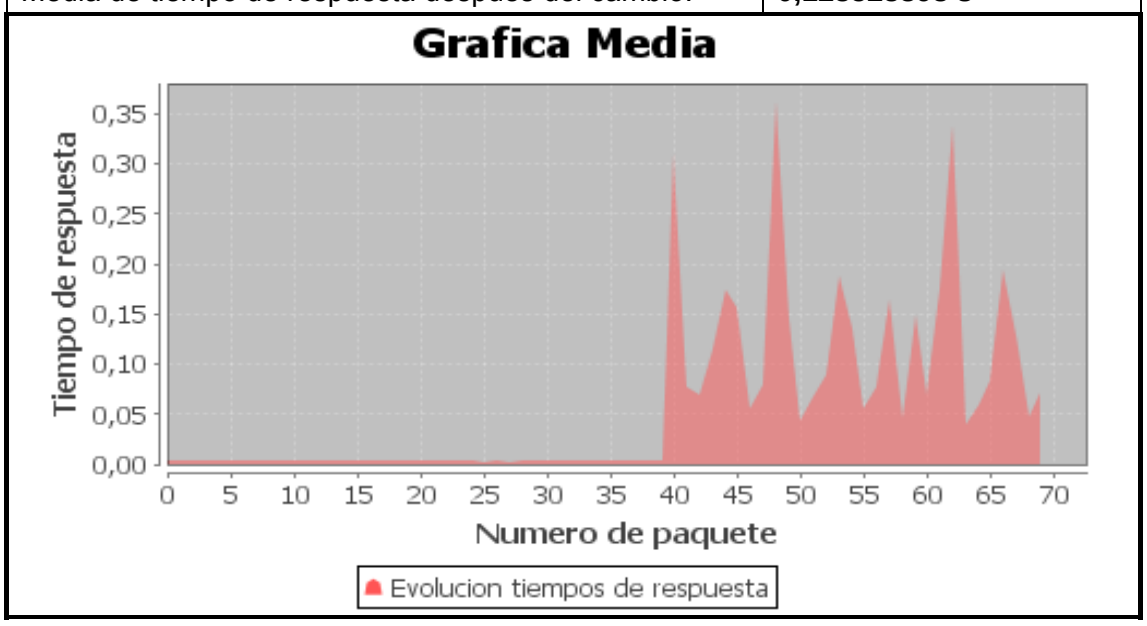
PRUEBA 1.1.2.1.D	
Número total de Request:	**120**
Número total de Reply:	**70**
Número de peticiones perdidas:	**50**
El cambio de red lo efectuó en el paquete:	**40**
Media de tiempo de respuesta antes del cambio:	**0.0026272093 s**
Media de tiempo de respuesta después del cambio:	**0.12533595 s**



PRUEBA 1.1.2.1.E	
Número total de Request:	**120**
Número total de Reply:	**89**
Número de peticiones perdidas:	**31**
El cambio de red lo efectuó en el paquete:	**40**
Media de tiempo de respuesta antes del cambio:	**0.0024846322 s**
Media de tiempo de respuesta después del cambio:	**0.11219803 s**



MEDIA DE LAS PRUEBA 1.1.2.1: PING NORMAL VPN ESTRUCTURA REAL	
El Número de peticiones perdidas:	**40,8**
Media de tiempo de respuesta antes del cambio:	**0,00250793916 s**
Media de tiempo de respuesta después del cambio:	**0,123523898 s**



Conclusiones

Primeramente, y como es obvio, se observa una diferencia notable entre el envío de paquetes en red local y el envío de paquetes a través de openvpn, por lo que ya se ha explicado anteriormente: los paquetes que se envían por openvpn deben ir primero al servidor openvpn, a través del bridge y del tap, añaden datos a los paquetes con medidas de seguridad… todo eso hace que el tiempo de recepción desde que se envía el paquete hasta que se recibe la confirmación de entrega (reply) aumente notablemente.

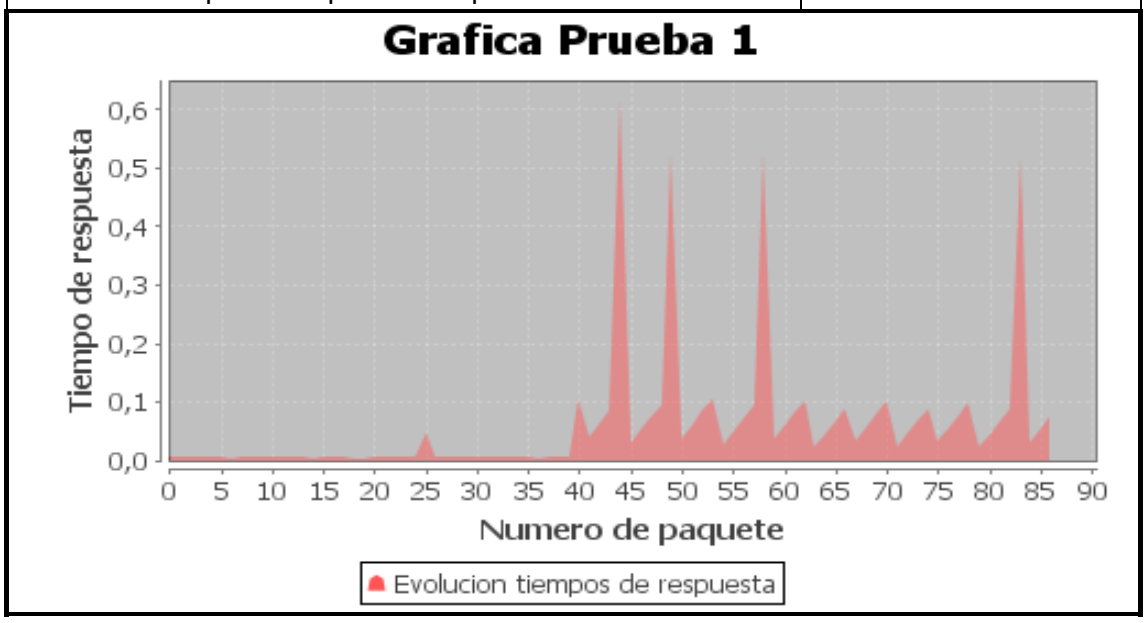
Otro dato interesante que en máquinas virtuales no se daba tanto es la irregularidad de los tiempos a través de openvpn, donde tenemos varios picos de tiempos grandes, y mucho más pronunciados. Esto se debe a que ya no estamos en una red ideal, sino en una real, y los paquetes ya no tardan lo mismo, sino que dependen del tráfico que haya en la red en ese momento, de la capacidad de las tarjetas de red, de la calidad de los routers, etc.

El tiempo que tarda en establecerse la conexión desde que se inicia el cambio es de unos 40.8 segundos. Este tiempo incluye: la conexión por wifi de una red a otra, lo cual requiere un tiempo de autentificación en el router; el tiempo de inicialización de openvpn; y por último, el tiempo que tarda la red en darse cuenta de que ya se han inicializado todos los procesos necesarios para que la comunicación continúe.

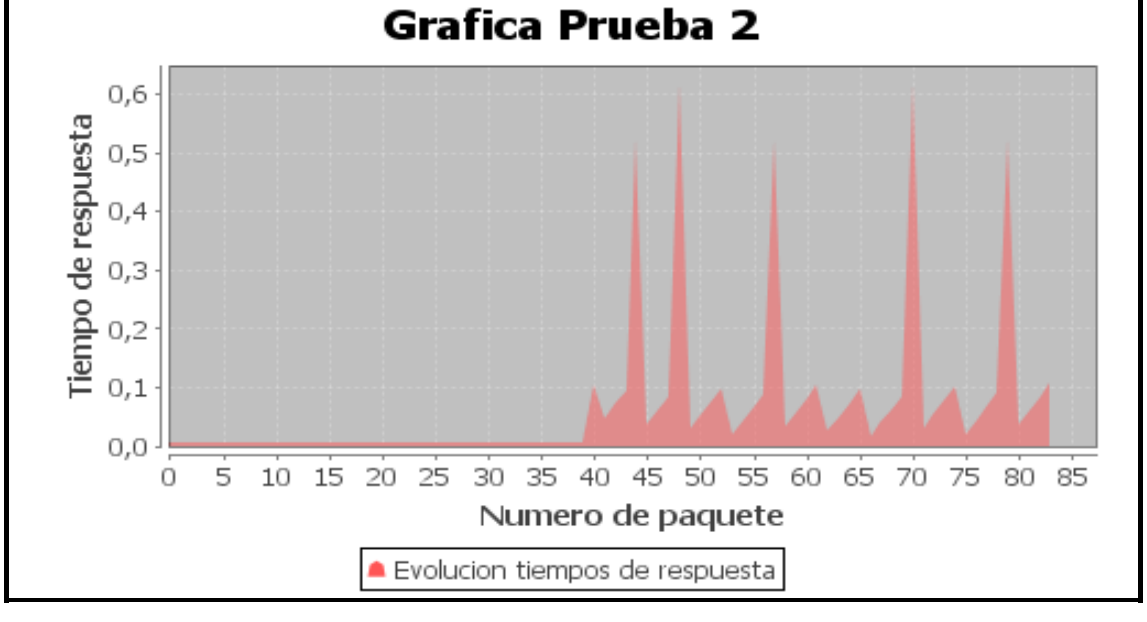
1.1.2.2 Envío de paquetes ping de 1000 bytes durante el cambio de red

Se comprueba el comportamiento de openvpn en una estructura real con un tamaño de datos mucho mayor, subiendo de 64 bytes a 1000 bytes. La frecuencia del envío de paquetes se sigue manteniendo en 1 paquete cada segundo.

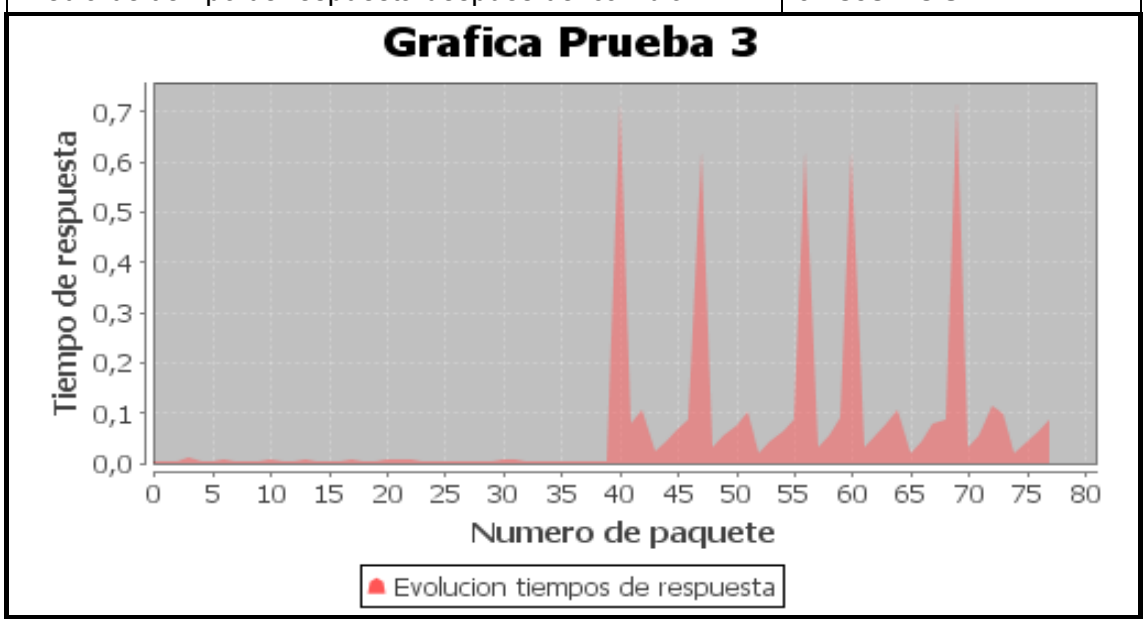
PRUEBA 1.1.2.2.A	
Número total de Request:	**120**
Número total de Reply:	**87**
Número de peticiones perdidas:	**33**
El cambio de red lo efectuó en el paquete:	**40**
Media de tiempo de respuesta antes del cambio:	**0.0051693497 s**
Media de tiempo de respuesta después del cambio:	**0.102942936 s**



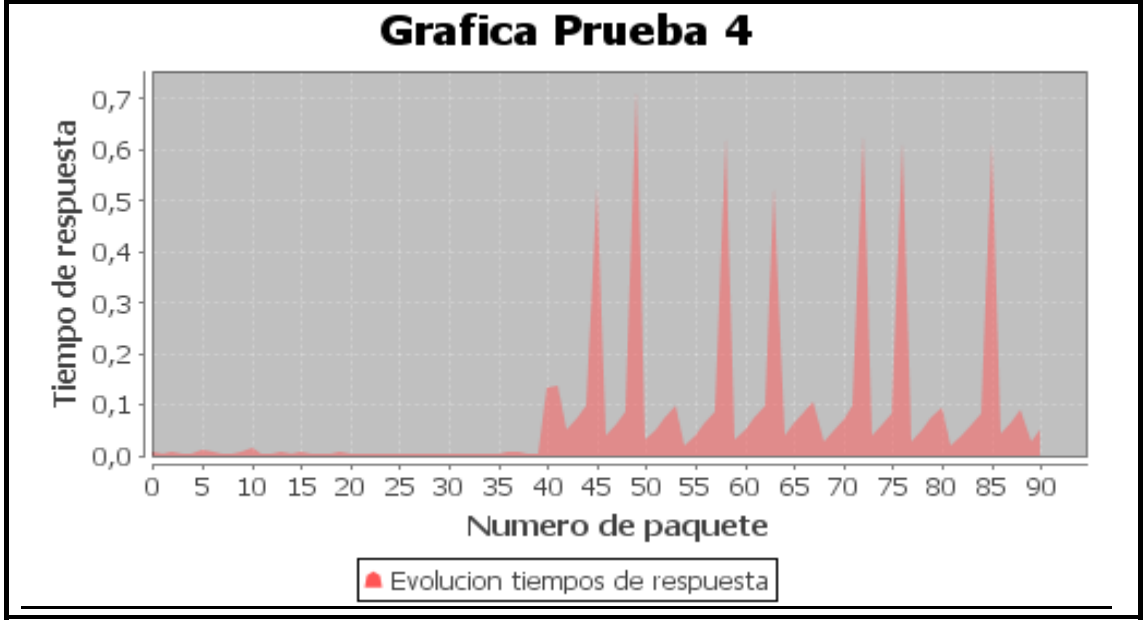
PRUEBA 1.1.2.2.B	
Número total de Request:	**120**
Número total de Reply:	**84**
Número de peticiones perdidas:	**36**
El cambio de red lo efectuó en el paquete:	**40**
Media de tiempo de respuesta antes del cambio:	**0.0042029605 s**
Media de tiempo de respuesta después del cambio:	**0.11862269 s**



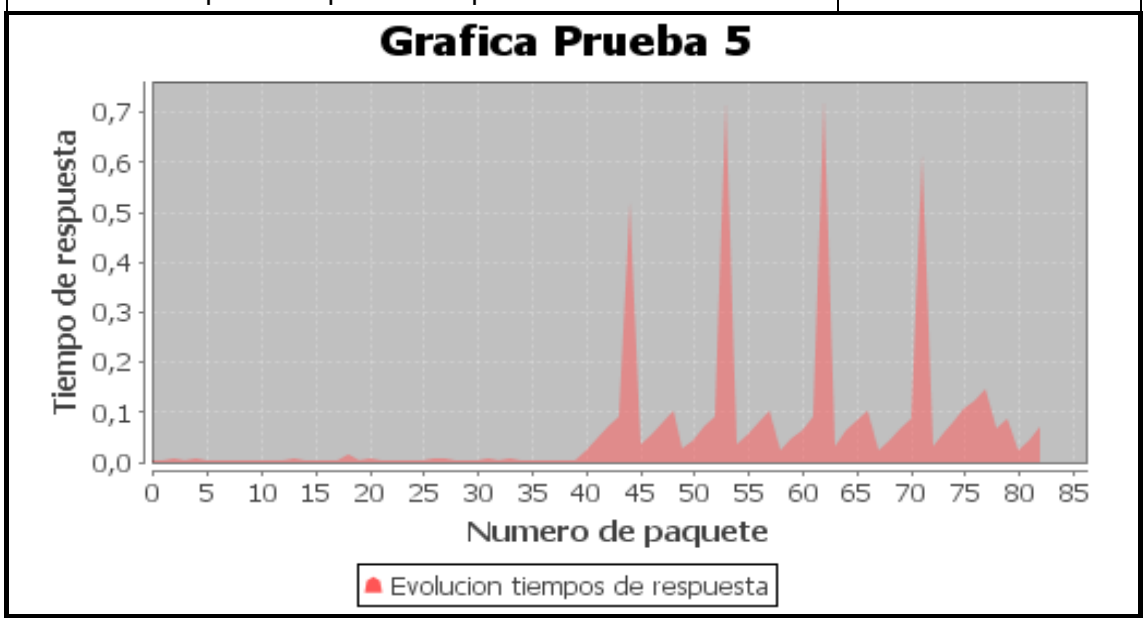
PRUEBA 1.1.2.2.C	
Número total de Request:	**120**
Número total de Reply:	**78**
Número de peticiones perdidas:	**42**
El cambio de red lo efectuó en el paquete:	**40**
Media de tiempo de respuesta antes del cambio:	**0.0044300826 s**
Media de tiempo de respuesta después del cambio:	**0.1398713 s**



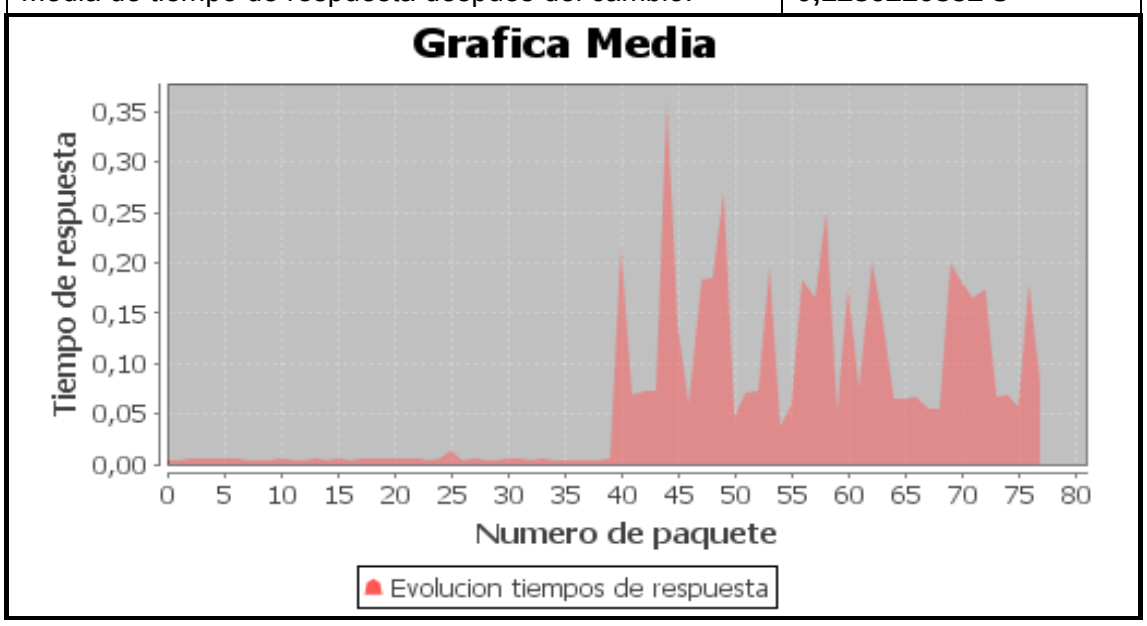
PRUEBA 1.1.2.2.D	
Número total de Request:	**121**
Número total de Reply:	**91**
Número de peticiones perdidas:	**30**
El cambio de red lo efectuó en el paquete:	**40**
Media de tiempo de respuesta antes del cambio:	**0.0047184704 s**
Media de tiempo de respuesta después del cambio:	**0.1378488 s**



PRUEBA 1.1.2.2.D	
Número total de Request:	**120**
Número total de Reply:	**83**
Número de peticiones perdidas:	**37**
El cambio de red lo efectuó en el paquete:	**40**
Media de tiempo de respuesta antes del cambio:	**0.0044641616 s**
Media de tiempo de respuesta después del cambio:	**0.11882445 s**



MEDIA DE LAS PRUEBA 1.1.2.2: PING 1000 bytes VPN ESTRUCTURA REAL	
El Número de peticiones perdidas:	**35.6**
Media de tiempo de respuesta antes del cambio:	**0,00459700496 s**
Media de tiempo de respuesta después del cambio:	**0,1236220352 s**



Conclusiones

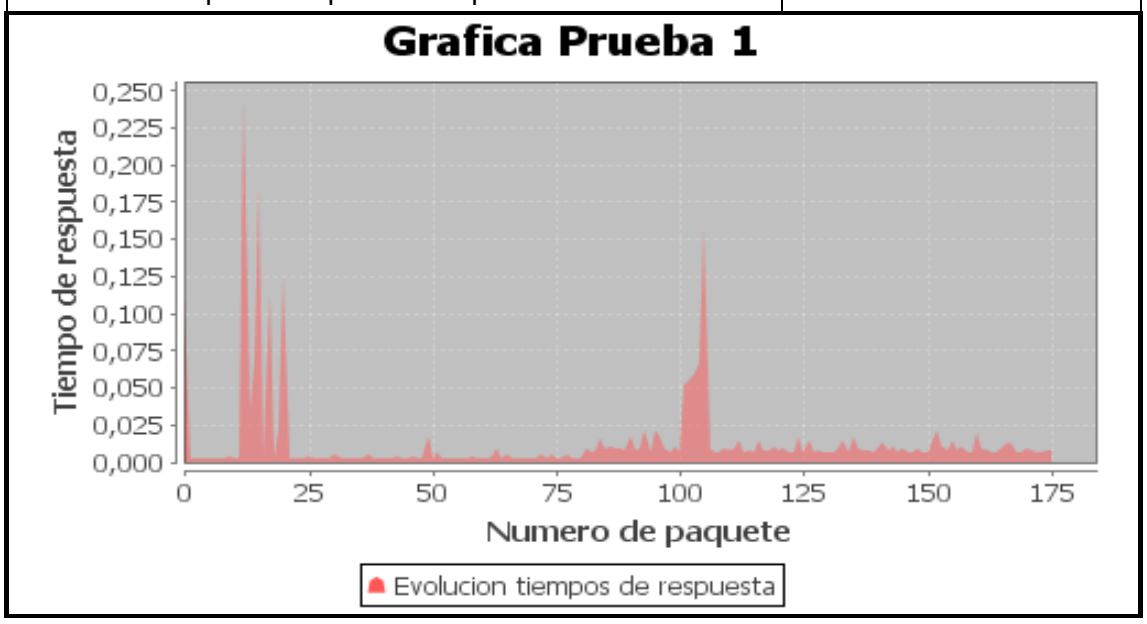
Al igual que ocurría en las máquinas virtuales, el número de peticiones que se pierden durante la espera es menor, por lo que el tiempo que tardará la conexión en establecerse a la nueva red para que la comunicación siga su curso también es menor, aunque, como ya se ha explicado, son datos tan sumamente pequeños que no tienen casi influencia. La irregularidad en el envío de paquetes cuando el cliente se traslada a otra red y se conecta por openvpn sigue estando, y en este ejemplo también es mucho más pronunciada que en el caso de las máquinas virtuales.

En cuanto a la relación de tiempos en ambas redes entre el envío de paquetes de 64 bytes y el de 1000 bytes, se puede ver que el tiempo en la red local pasa de 0,00250793916 a 0,00459700496 s, lo que supone un incremento del 83% del tiempo, un dato bastante importante. Respecto a la red a través de openvpn, el tiempo pasa de 0,123523898 a 0,1236220352 s, cuyo aumento es casi nulo. Por lo tanto, se puede deducir que con el incremento del tamaño de los paquetes, la red de openvpn a penas notará diferencia mientras la red local sufrirá una subida de tiempo importante, aunque obviamente siempre será mejor la red local que una red externa conectada a través de openvpn.

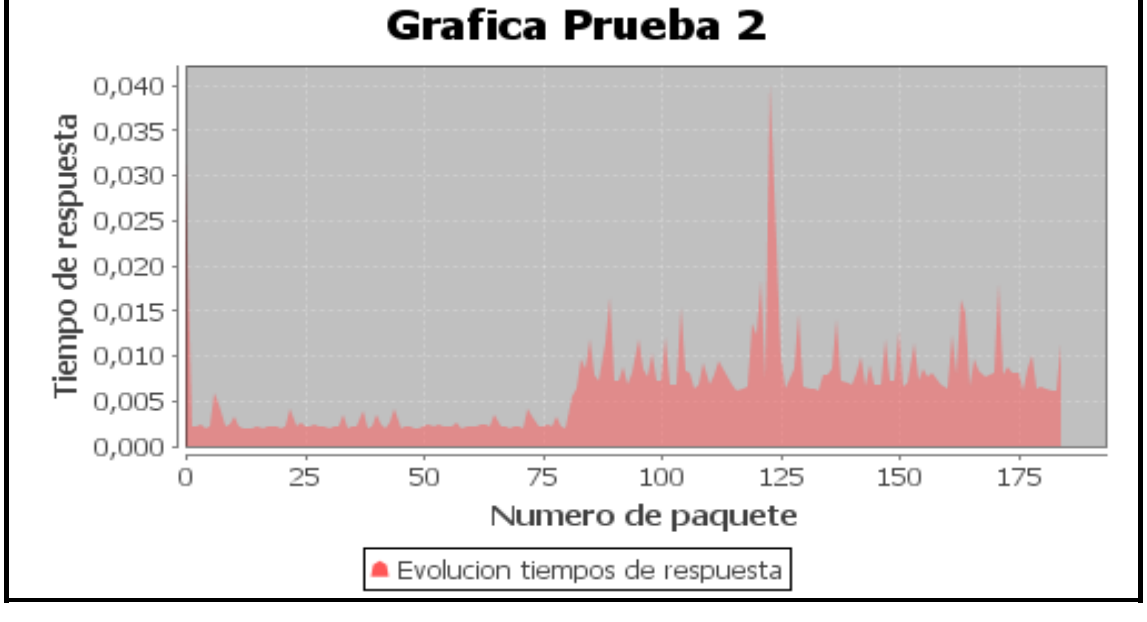
1.1.2.3 Envío de paquetes ping cada 0.5 segundos durante el cambio de red

En la siguiente prueba se estudia el comportamiento de openvpn en una estructura real cuando el envío de datos se hace a una frecuencia mayor, concretamente cuando se tiene una frecuencia de 2 hz (un paquete cada 0.5 segundos).

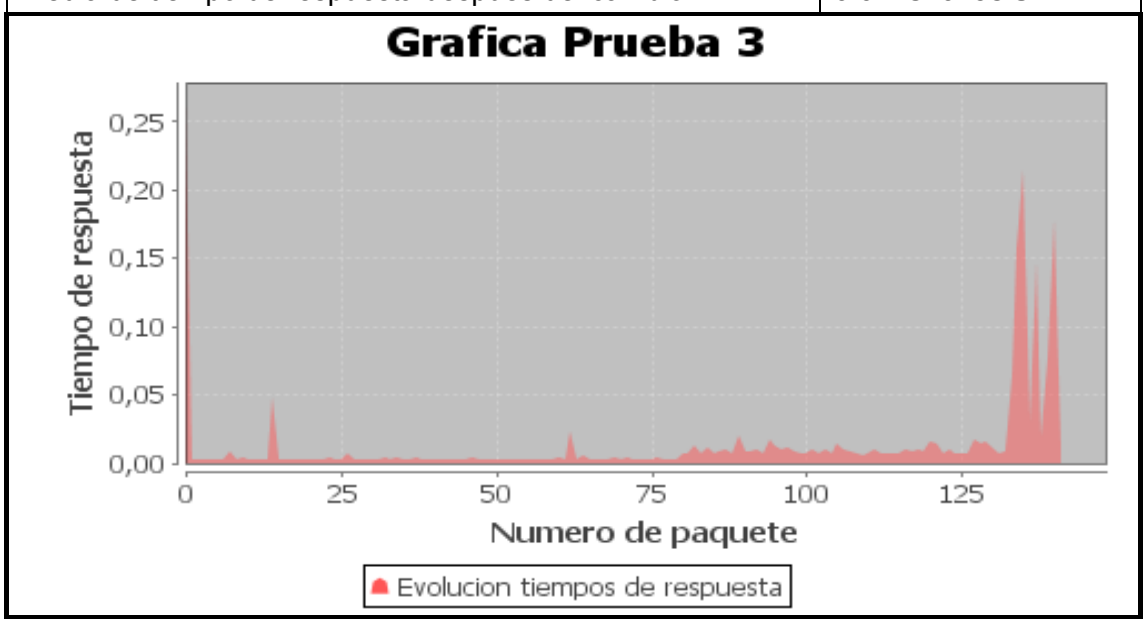
PRUEBA 1.1.2.3.A	
Número total de Request:	**240**
Número total de Reply:	**176**
Número de peticiones perdidas:	**64**
El cambio de red lo efectuó en el paquete:	**81**
Media de tiempo de respuesta antes del cambio:	**0.013651777 s**
Media de tiempo de respuesta después del cambio:	**0.012914477 s**



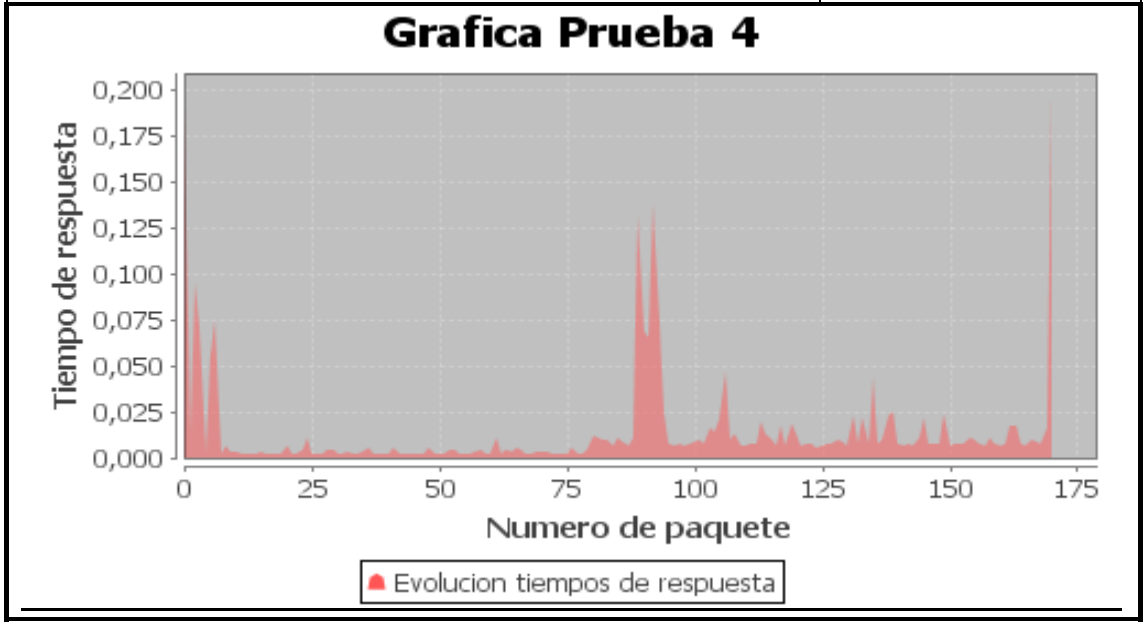
PRUEBA 1.1.2.3.B	
Número total de Request:	**240**
Número total de Reply:	**185**
Número de peticiones perdidas:	**55**
El cambio de red lo efectuó en el paquete:	**81**
Media de tiempo de respuesta antes del cambio:	**0.0027351358 s**
Media de tiempo de respuesta después del cambio:	**0.009070249 s**



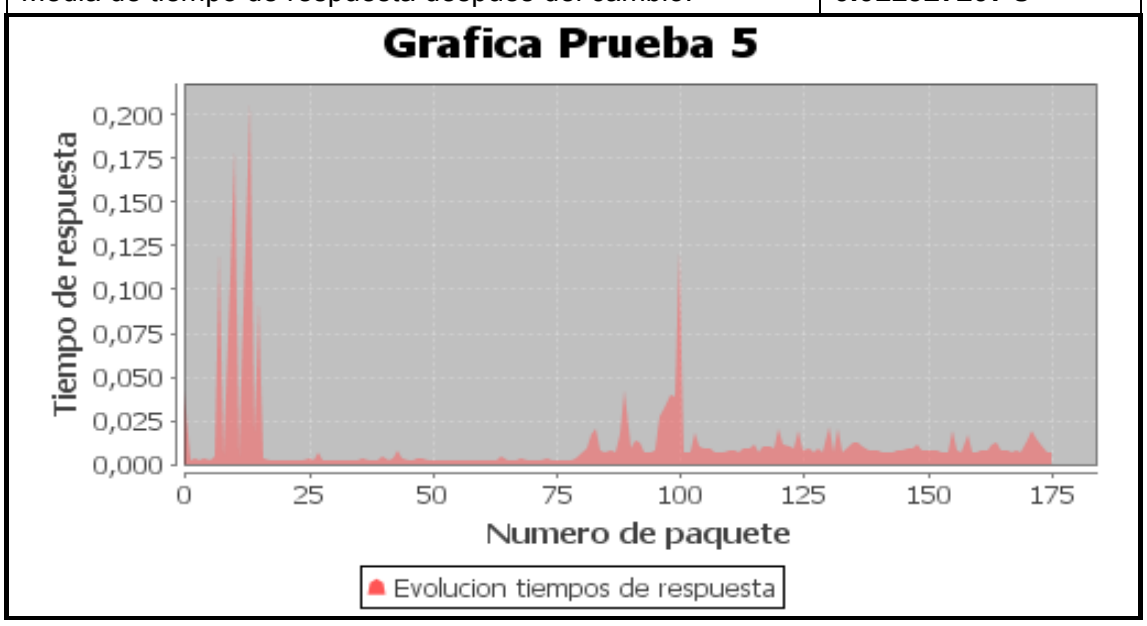
PRUEBA 1.1.2.3.C	
Número total de Request:	**240**
Número total de Reply:	**142**
Número de peticiones perdidas:	**98**
El cambio de red lo efectuó en el paquete:	**80**
Media de tiempo de respuesta antes del cambio:	**0.006566763 s**
Media de tiempo de respuesta después del cambio:	**0.022310195 s**



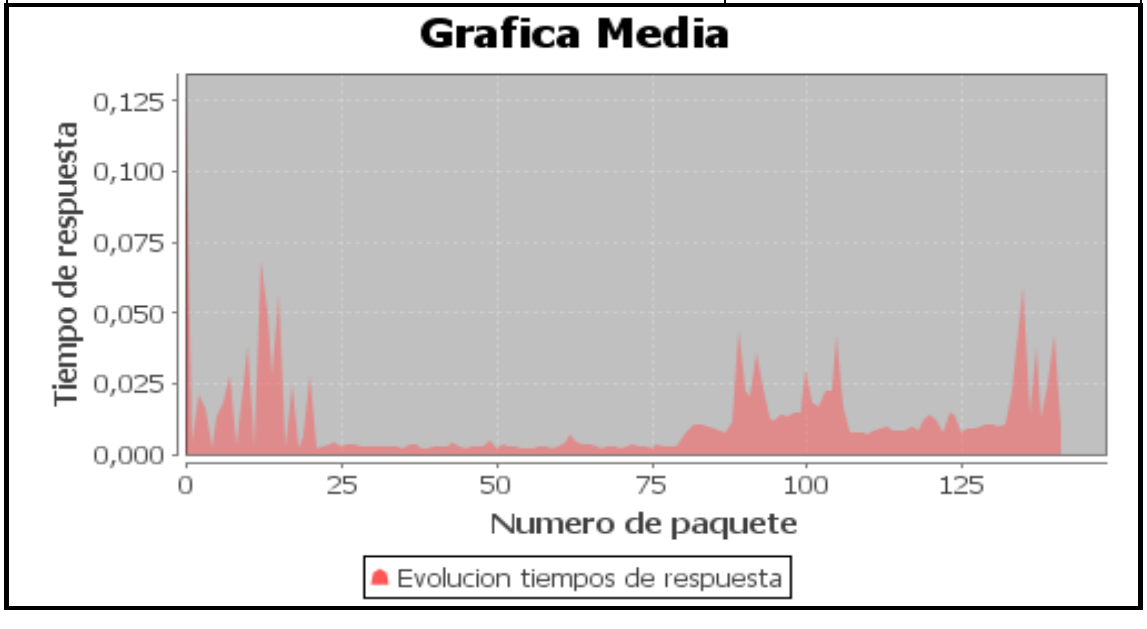
PRUEBA 1.1.2.3.D	
Número total de Request:	**240**
Número total de Reply:	**171**
Número de peticiones perdidas:	**69**
El cambio de red lo efectuó en el paquete:	**79**
Media de tiempo de respuesta antes del cambio:	**0.0090017775 s**
Media de tiempo de respuesta después del cambio:	**0.011184441 s**



PRUEBA 1.1.2.3.E	
Número total de Request:	**240**
Número total de Reply:	**176**
Número de peticiones perdidas:	**64**
El cambio de red lo efectuó en el paquete:	**80**
Media de tiempo de respuesta antes del cambio:	**0.012854266 s**
Media de tiempo de respuesta después del cambio:	**0.011927207 s**



MEDIA DE LAS PRUEBA 1.1.2.3: PING 0.5 seg VPN ESTRUCTURA REAL	
El Número de peticiones perdidas:	**70**
Media de tiempo de respuesta antes del cambio:	**0,00896194386 s**
Media de tiempo de respuesta después del cambio:	**0,0147571482 s**



Conclusiones

Cabe destacar en esta prueba que se obtuvieron unos picos muy altos al empezar el envío de paquetes en la red local, sobre todo en las pruebas 1 y 5, lo que hacía subir mucho la media de recepción en red local, pero si se observan las gráficas, se ve que el resto de paquetes, a excepción de esos 3 o 4, son igual de pequeños que antes.

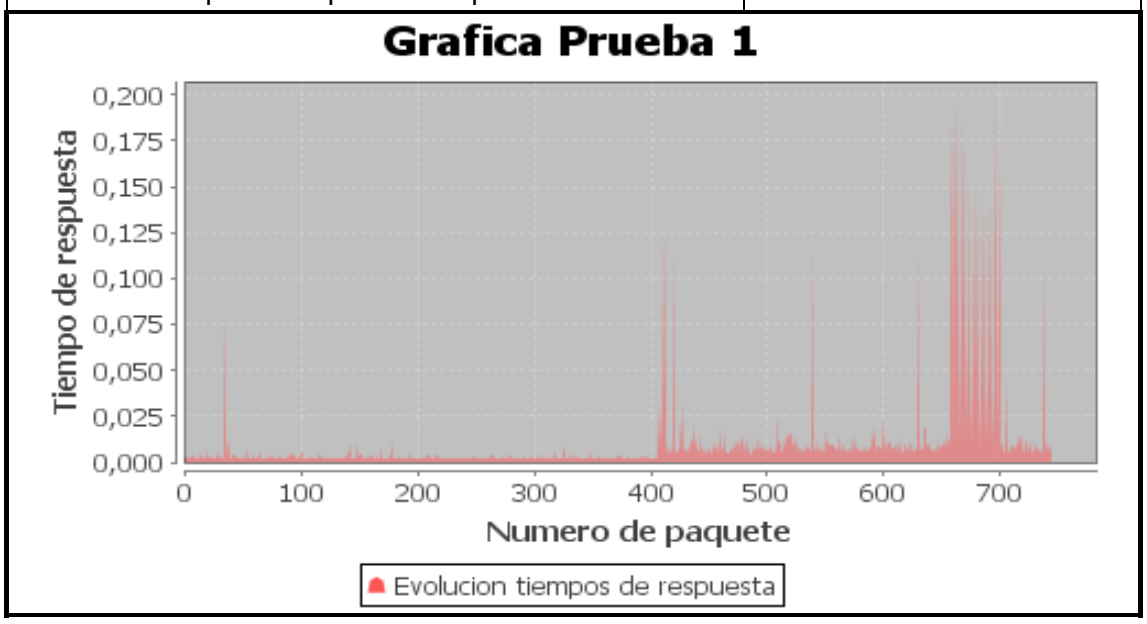
Se puede observar que el número de paquetes perdidos es 70, lo cual, comparado con el ejemplo básico de ping, hace que haya disminuido un poco la media de paquetes perdidos. Esto se debe a que, al tener más frecuencia de envío de paquetes, se puede afinar un poco más este aspecto. Así, el tiempo que pasa hasta volver a establecerse la conexión, al tener una frecuencia de 0.5, es de unos 35 segundos.

Respecto a los tiempos medios en ambas redes, como era de esperar, sigue siendo más alto en la red a través de openvpn que en la propia red local, concretamente un 64%, procedente de todo el tráfico de datos que trae consigo openvpn, tanto de redireccionamiento de datos a través del tap y bridge contra el servidor, como con motivos de autentificación y seguridad.

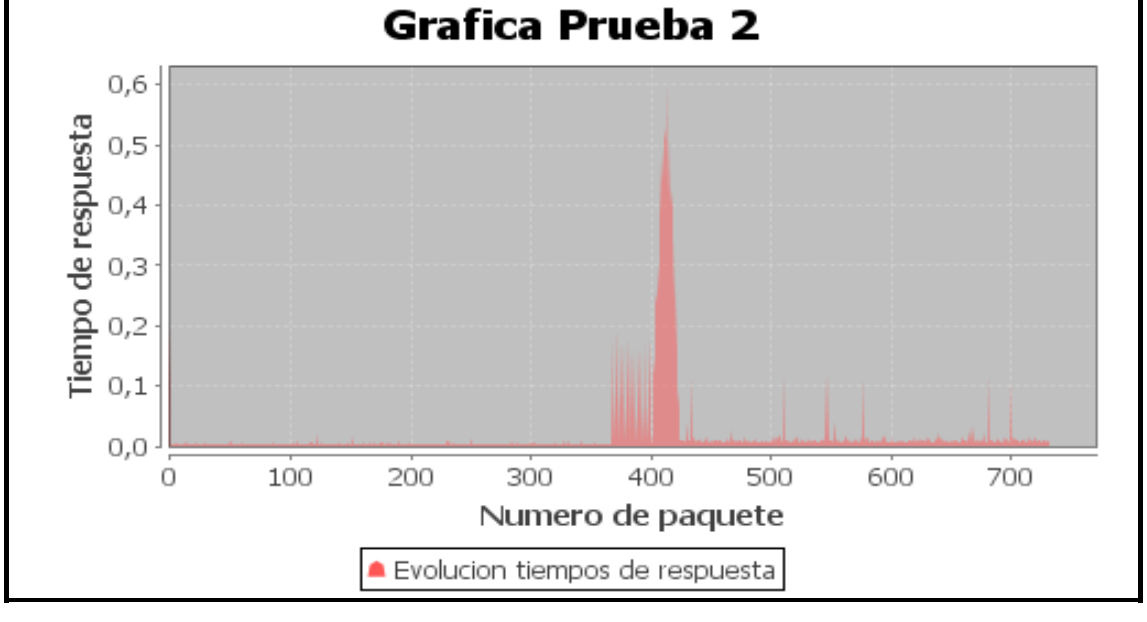
1.1.2.4 Envío de paquetes ping cada 0.1 segundos durante el cambio de red

En la siguiente prueba se estudia el comportamiento de openvpn en una estructura real cuando el envío de datos se hace a una frecuencia mayor, concretamente cuando se tiene una frecuencia de 10 hz (un paquete cada 0.1 segundos).

PRUEBA 1.1.2.4.A	
Número total de Request:	**987**
Número total de Reply:	**746**
Número de peticiones perdidas:	**241**
El cambio de red lo efectuó en el paquete:	**407**
Media de tiempo de respuesta antes del cambio:	**0.002719541 s**
Media de tiempo de respuesta después del cambio:	**0.018811364 s**

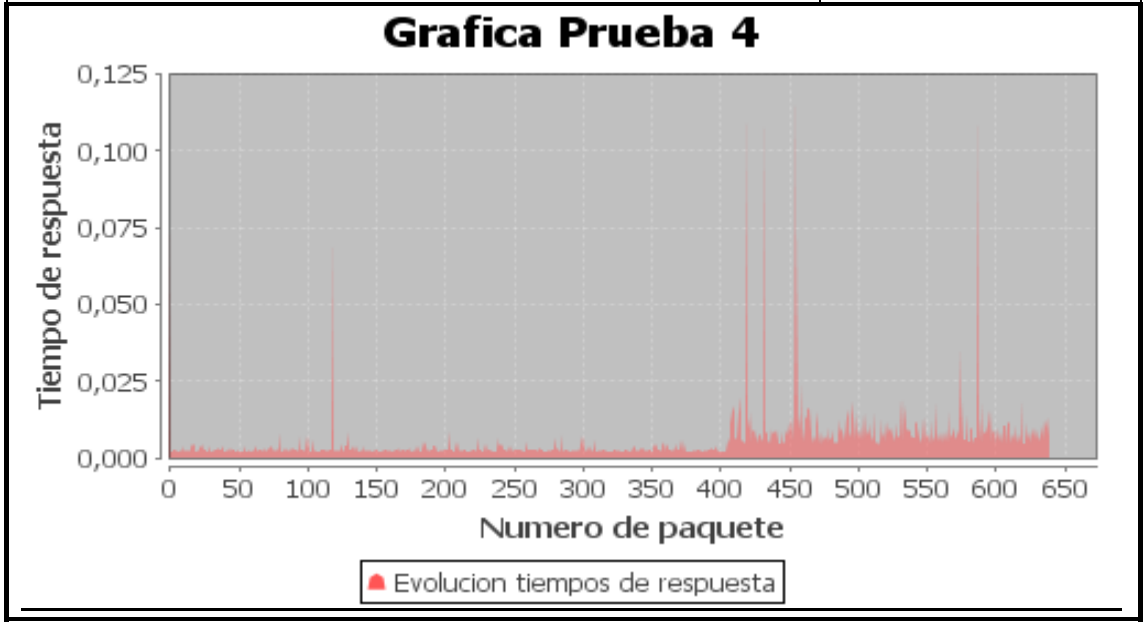


PRUEBA 1.1.2.4.B	
Número total de Request:	**978**
Número total de Reply:	**734**
Número de peticiones perdidas:	**244**
El cambio de red lo efectuó en el paquete:	**402**
Media de tiempo de respuesta antes del cambio:	**0.009743159 s**
Media de tiempo de respuesta después del cambio:	**0.03184962 s**

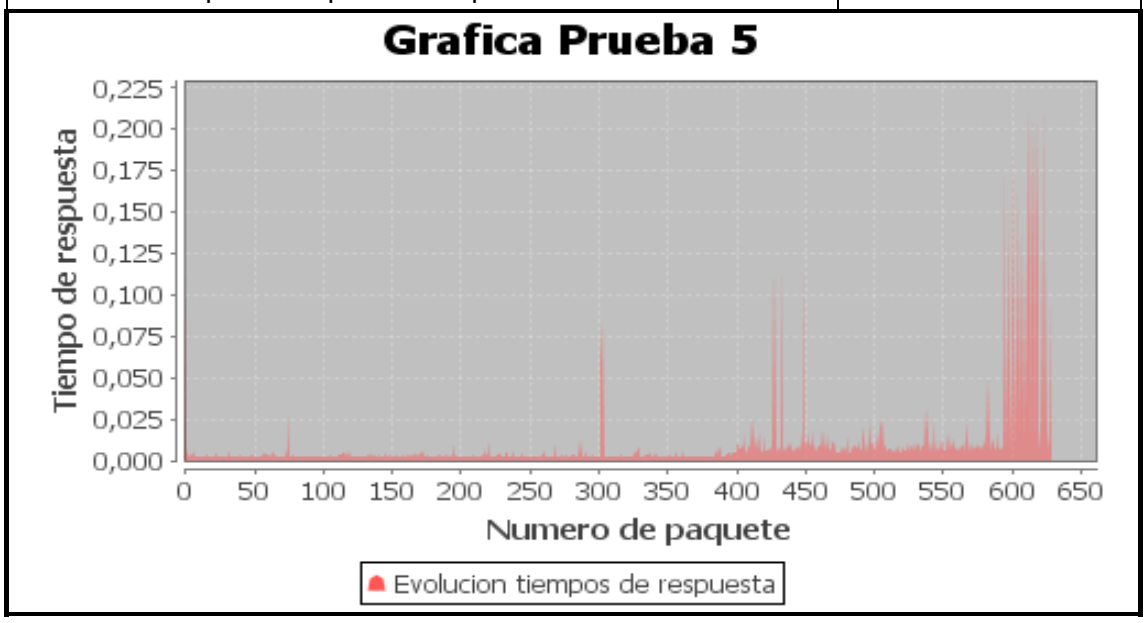


PRUEBA 1.1.2.4.C	
Número total de Request:	**995**
Número total de Reply:	**749**
Número de peticiones perdidas:	**246**
El cambio de red lo efectuó en el paquete:	**401**
Media de tiempo de respuesta antes del cambio:	**0.0028079976 s**
Media de tiempo de respuesta después del cambio:	**0.011078473 s**

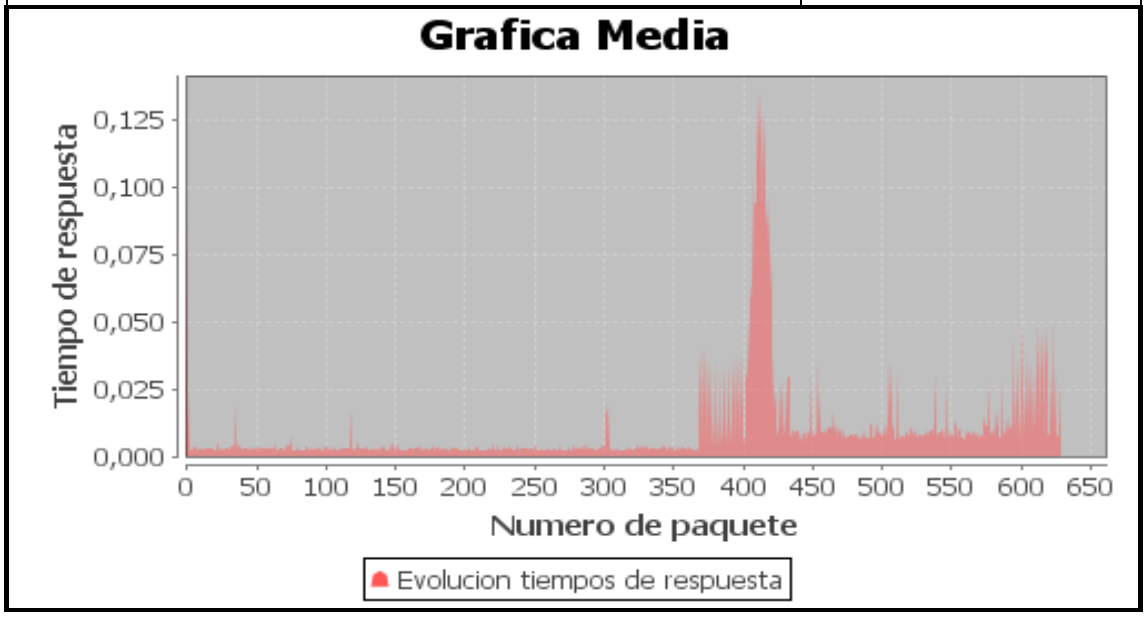
PRUEBA 1.1.2.4.D	
Número total de Request:	**997**
Número total de Reply:	**641**
Número de peticiones perdidas:	**356**
El cambio de red lo efectuó en el paquete:	**405**
Media de tiempo de respuesta antes del cambio:	**0.0028471826 s**
Media de tiempo de respuesta después del cambio:	**0.010474739 s**



PRUEBA 1.1.2.4.E	
Número total de Request:	**1000**
Número total de Reply:	**630**
Número de peticiones perdidas:	**370**
El cambio de red lo efectuó en el paquete:	**400**
Media de tiempo de respuesta antes del cambio:	**0.0032902793 s**
Media de tiempo de respuesta después del cambio:	**0.021462647 s**



MEDIA DE LAS PRUEBA 1.1.2.4: PING 0.1 seg VPN ESTRUCTURA REAL	
El Número de peticiones perdidas:	**291,4**
Media de tiempo de respuesta antes del cambio:	**0,00791610544 s**
Media de tiempo de respuesta después del cambio:	**0,0187353686 s**



Conclusiones

En esta última prueba sobre openvpn en una estructura real. Al igual que pasaba antes, el número de paquetes perdidos de media baja, y vemos que el tiempo medio que tarda la red en volver a conectarse es de 29.1 segundos. Ahora vemos que las pruebas tienen muchos menos picos que los que tenían en la anterior.

Además, al tener un tamaño de muestra mucho mayor, los picos que se puedan tener (como los que se aprecian sobre todo en la red openvpn) no son tan determinantes para la media, por lo que el valor de ésta tiene mayor relevancia.

El tiempo medio en ambas redes sigue con el patrón lógico y cumplido hasta ahora: con mayor tiempo medio en la red a través de openvpn que en la red en local. Ahora esta diferencia se encuentra en torno al 136%. Como ya se ha comentado, es normal que sea mayor, pero si en este caso lo es tanto es por los picos de tiempo que se hallan en la red openvpn que ya hemos comentado, los cuales hacen subir aún más ese tiempo, y acentúan la diferencia entre ambos.

1.2- HIP

Para HIP vamos a desarrollar las mismas pruebas que hemos desarrollado para openvpn, que fueron:

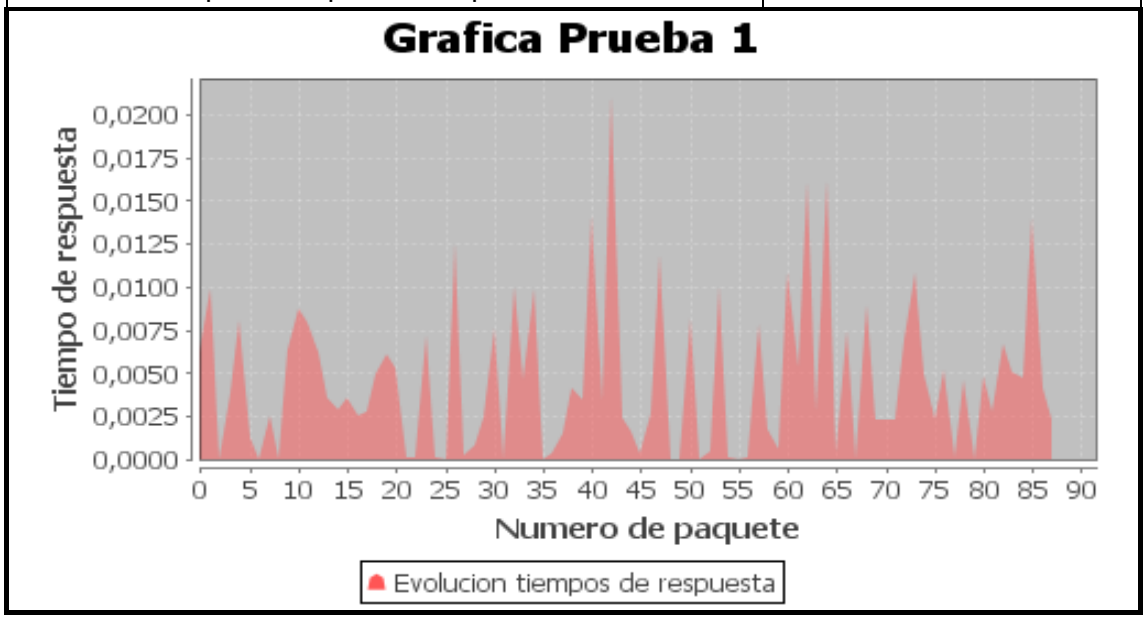
- Envío de paquetes ping básicos durante el cambio de red.
- Envío de paquetes ping de 1000 bytes durante el cambio de red.
- Envío de paquetes ping cada 0.5 segundos durante el cambio de red.
- Envío de paquetes ping cada 0.1 segundos durante el cambio de red.

💻 1.2.1- Virtual

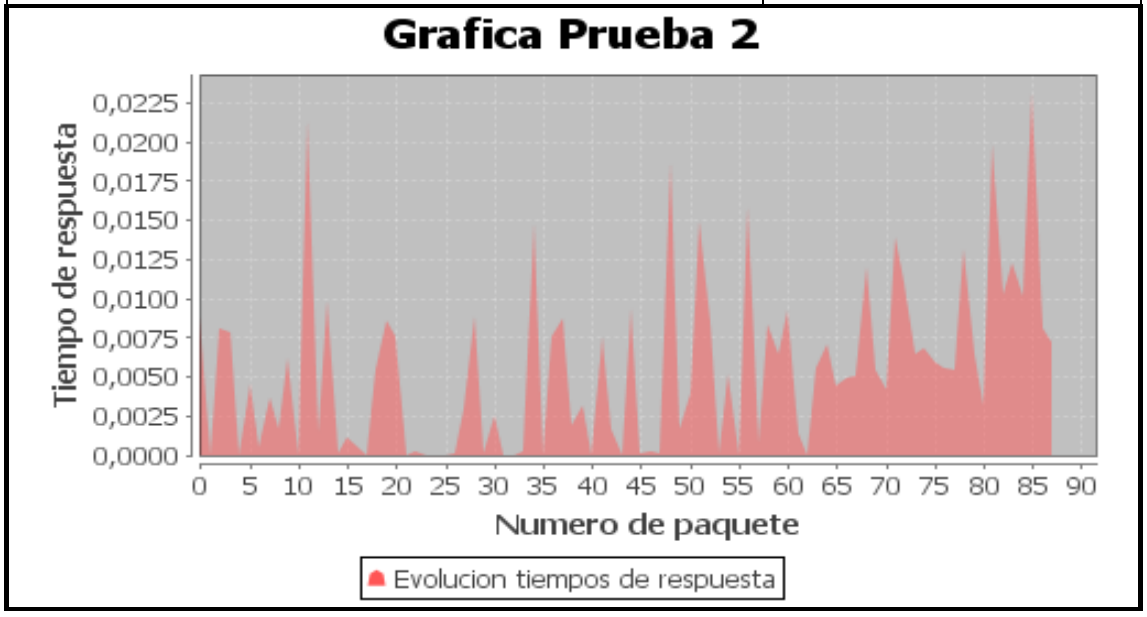
1.2.1.1 Envío de paquetes ping de básicos durante el cambio de red

La primera prueba a realizar es el ping básico, es decir, el envío de paquetes de 64 bytes cada segundo.

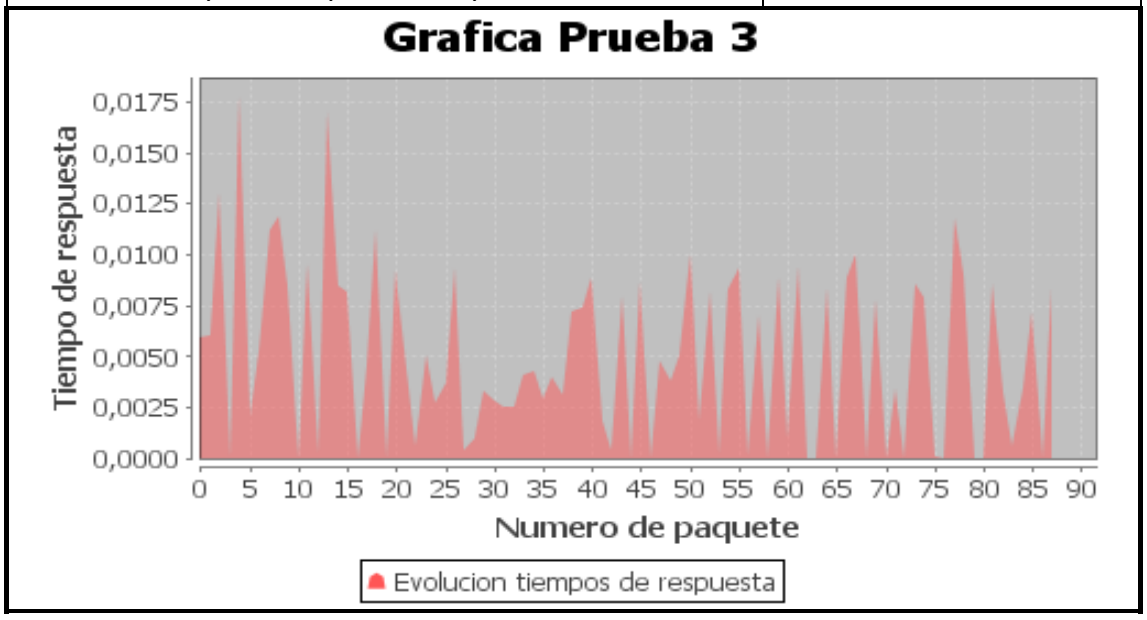
PRUEBA 1.2.1.1.A	
Número total de Request:	**100**
Número total de Reply:	**88**
Número de peticiones perdidas:	**12**
El cambio de red lo efectuó en el paquete:	**40**
Media de tiempo de respuesta antes del cambio:	**0.0039066514 s**
Media de tiempo de respuesta después del cambio:	**0.00498875 s**



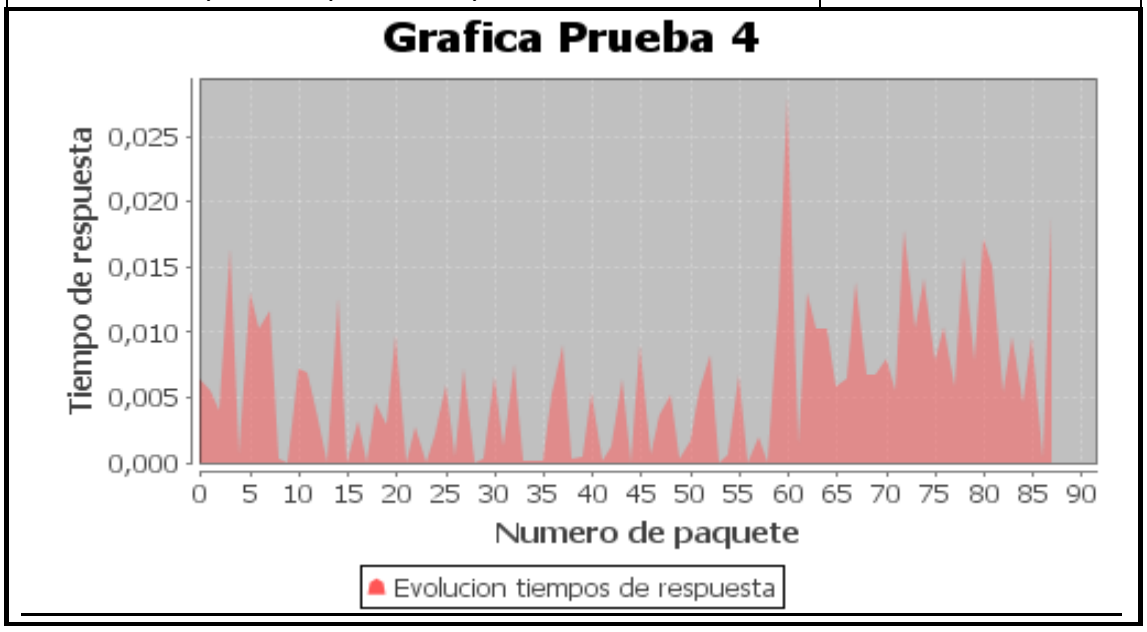
PRUEBA 1.2.1.1.B	
Número total de Request:	**100**
Número total de Reply:	**88**
Número de peticiones perdidas:	**12**
El cambio de red lo efectuó en el paquete:	**40**
Media de tiempo de respuesta antes del cambio:	**0.0036845445 s**
Media de tiempo de respuesta después del cambio:	**0.006881396 s**



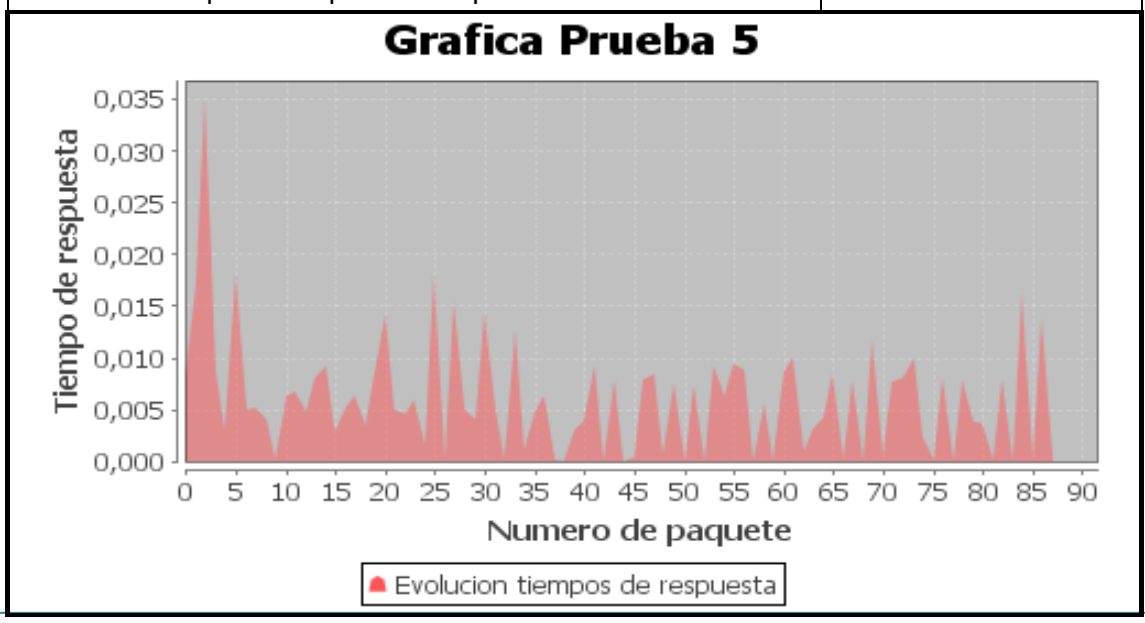
PRUEBA 1.2.1.1.C	
Número total de Request:	**100**
Número total de Reply:	**88**
Número de peticiones perdidas:	**12**
El cambio de red lo efectuó en el paquete:	**40**
Media de tiempo de respuesta antes del cambio:	**0.005505058 s**
Media de tiempo de respuesta después del cambio:	**0.004407724 s**



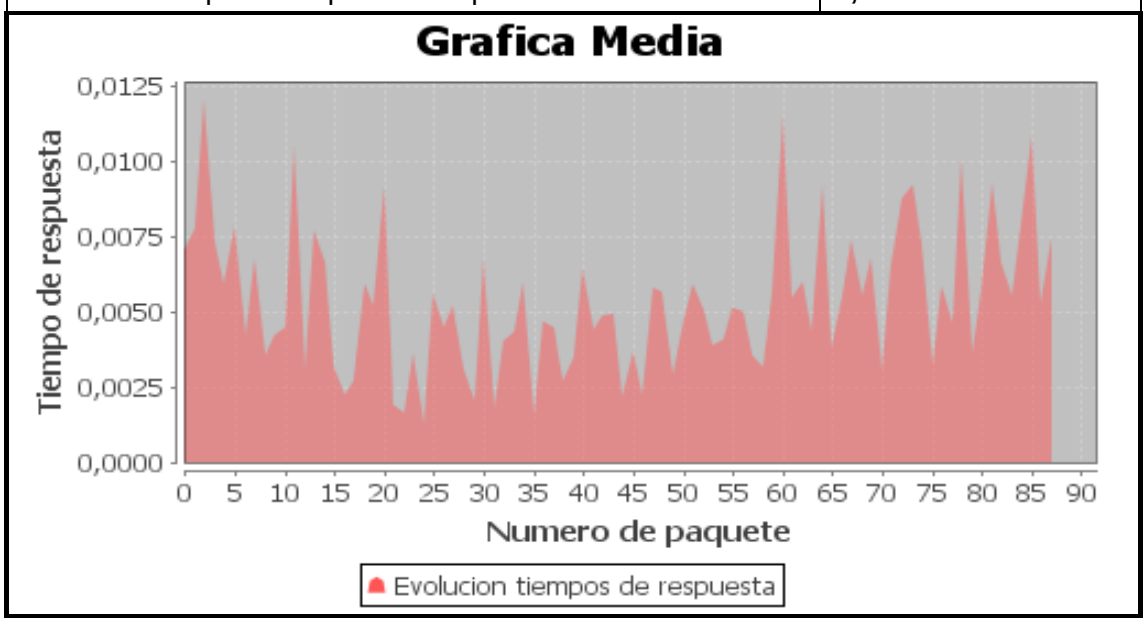
PRUEBA 1.2.1.1.D	
Número total de Request:	**100**
Número total de Reply:	**88**
Número de peticiones perdidas:	**12**
El cambio de red lo efectuó en el paquete:	**40**
Media de tiempo de respuesta antes del cambio:	**0.0041598724 s**
Media de tiempo de respuesta después del cambio:	**0.007355372 s**



PRUEBA 1.2.1.1.E	
Número total de Request:	**100**
Número total de Reply:	**88**
Número de peticiones perdidas:	**12**
El cambio de red lo efectuó en el paquete:	**40**
Media de tiempo de respuesta antes del cambio:	**0.0071325996 s**
Media de tiempo de respuesta después del cambio:	**0.0049139657 s**



MEDIA DE LAS PRUEBA 1.2.1.1: PING NORMAL HIP ESTRUCTURA VIRTUAL	
El Número de peticiones perdidas:	**12**
Media de tiempo de respuesta antes del cambio:	**0,00487774518 s**
Media de tiempo de respuesta después del cambio:	**0,00570944154 s**



Conclusiones

El número de paquetes perdidos de media es 12 (realmente, en todos los casos), por lo que, a 1 segundo por paquete, tenemos que la conexión queda interrumpida durante 12 segundos, después de los cuales, ya se restaura la conexión.

Un detalle que se observa es que hay muchos picos, no es para nada lineal. Algunos tienen valores mínimos, casi insignificantes, y otros suben muy por encima de la media. Eso da a entender que HIP tiene unos valores menos homogéneos de los obtenidos en openvpn, pero obtiene mejores resultados, pues son más rápidos.

Otro detalle muy curioso es que en las pruebas 3 y 5, la media del tiempo en la red local es superior a la media del tiempo en la red foránea a través de HIP. Esto no es muy lógico, y se debe a dos factores, que son: que el tiempo en red local y el tiempo a través de hip es muy parejo, un poco superior el segundo; y a que en la red local en estas pruebas se han registrado unos picos altos que han hecho subir la media, por eso se obtienen estos resultados.

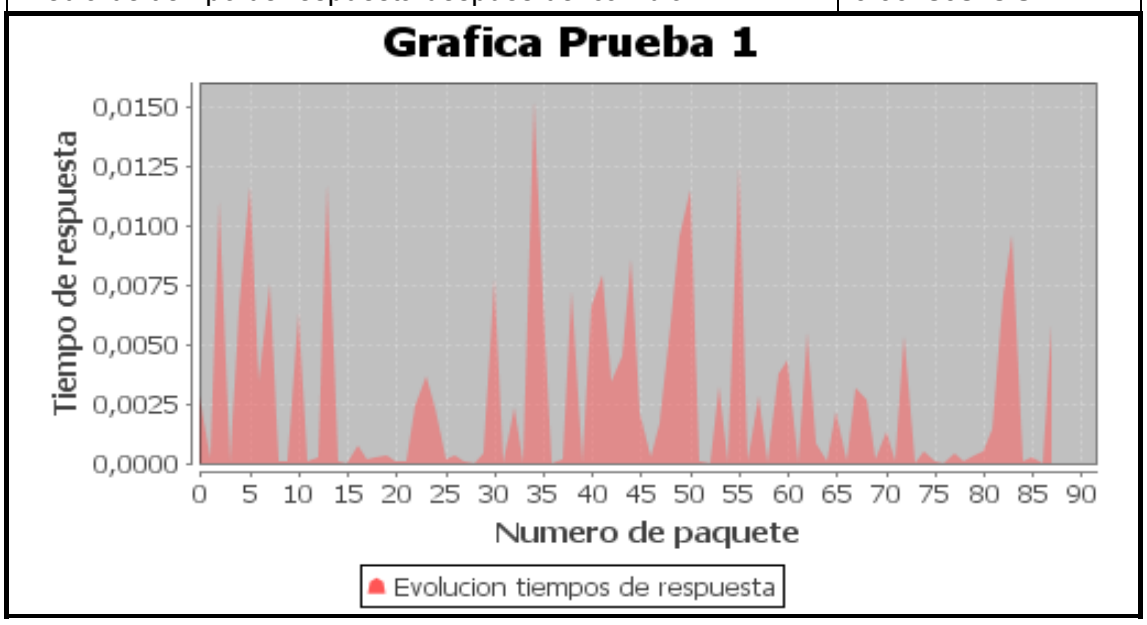
El tiempo de estar en red local o a través de HIP es similar, sube 0.00097 segundos, lo cual es un muy buen resultado. Esta subida se corresponde con un 19%, que es una subida menor de la que se obtenía cuando la movilidad estudiada era openvpn.

1.2.1.2 Envío de paquetes ping de 1000 bytes durante el cambio de red

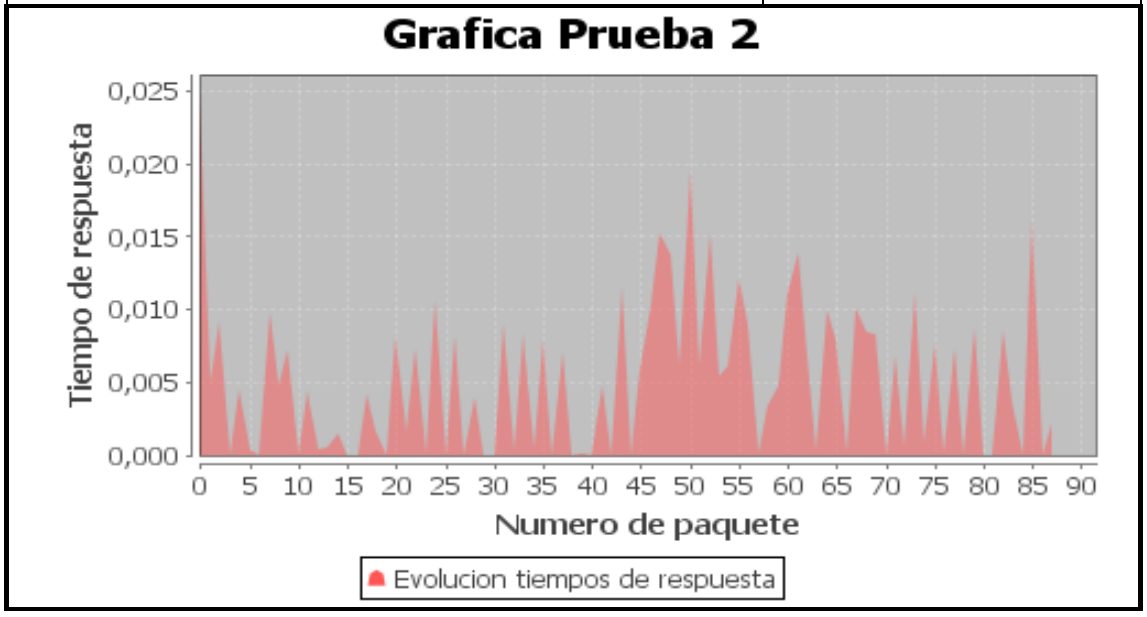
Al igual que en openvpn, una prueba a realizar fue aumentar el tamaño de los paquetes que se enviaban, pasando a tener un valor de 1000 bytes, y ver el comportamiento de HIP con una carga de trabajo de estas características.

A continuación se adjuntan las 5 pruebas realizadas para este estudio con su gráfica y los datos de la media de las 5 pruebas.

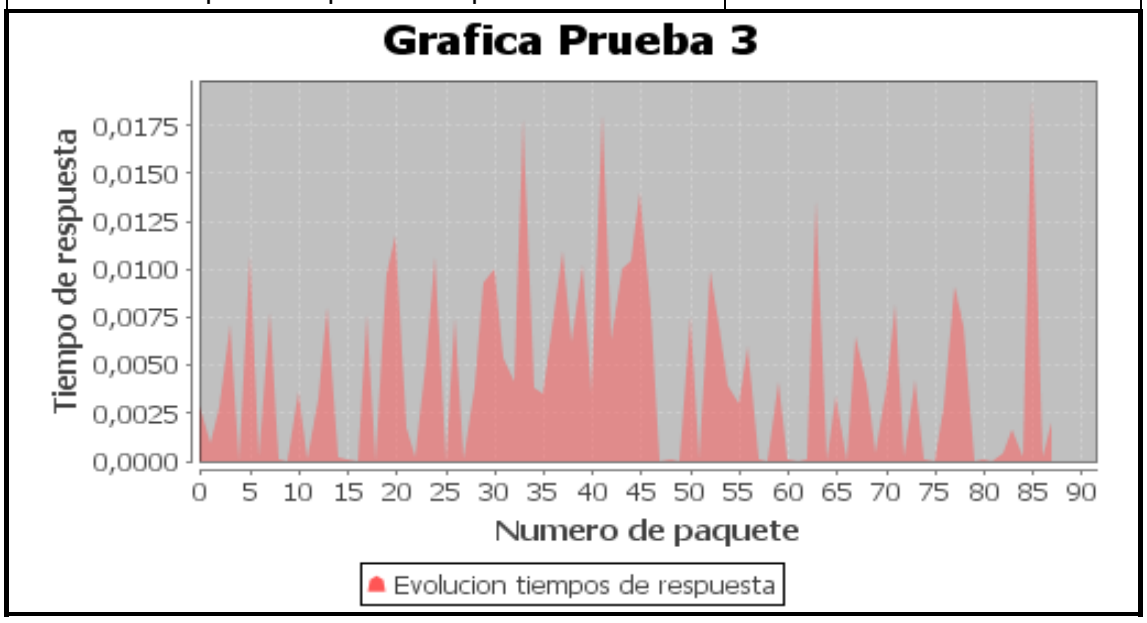
PRUEBA 1.2.1.2.A	
Número total de Request:	**100**
Número total de Reply:	**88**
Número de peticiones perdidas:	**12**
El cambio de red lo efectuó en el paquete:	**40**
Media de tiempo de respuesta antes del cambio:	**0.0027900296 s**
Media de tiempo de respuesta después del cambio:	**0.00280873 s**



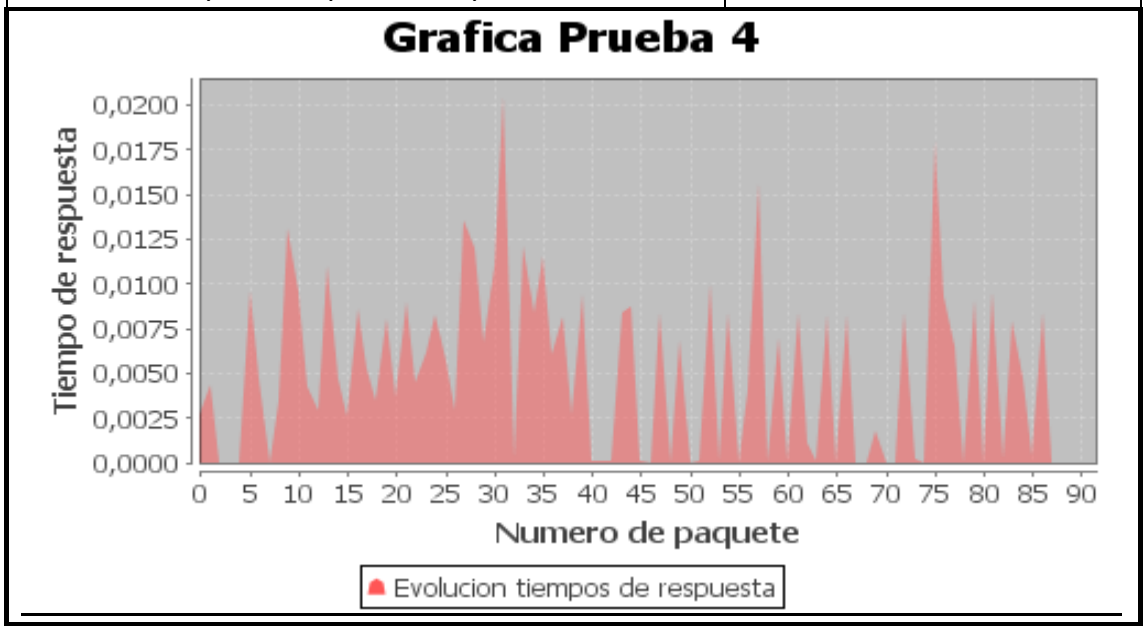
PRUEBA 1.2.1.2.B	
Número total de Request:	**100**
Número total de Reply:	**88**
Número de peticiones perdidas:	**12**
El cambio de red lo efectuó en el paquete:	**40**
Media de tiempo de respuesta antes del cambio:	**0.0037624657 s**
Media de tiempo de respuesta después del cambio:	**0.0062110424 s**



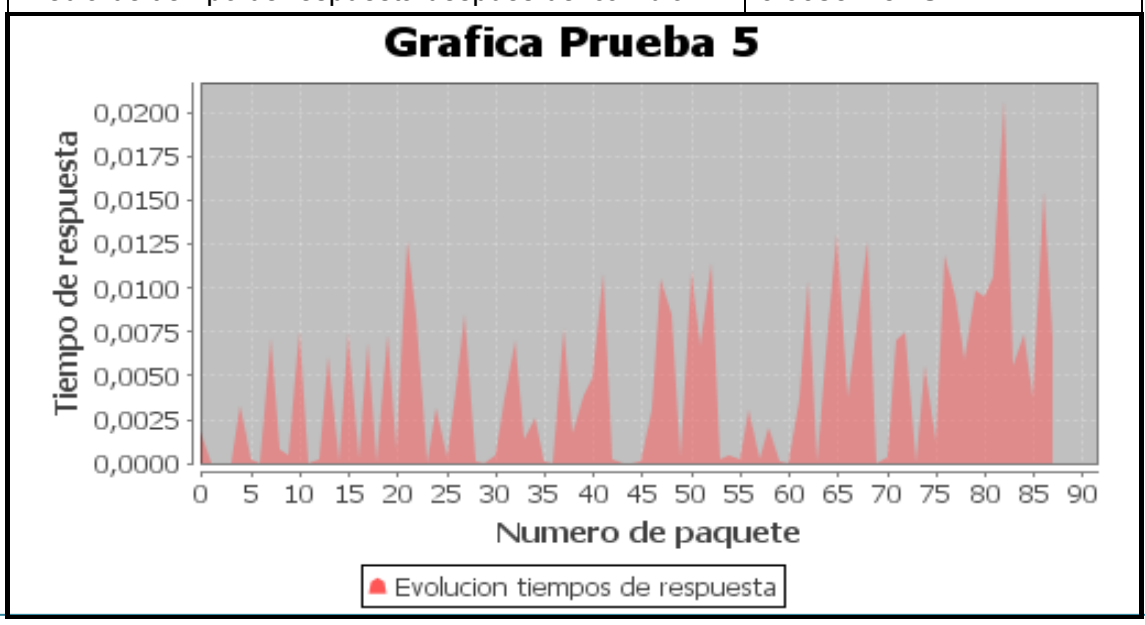
PRUEBA 1.2.1.2.C	
Número total de Request:	**100**
Número total de Reply:	**88**
Número de peticiones perdidas:	**12**
El cambio de red lo efectuó en el paquete:	**40**
Media de tiempo de respuesta antes del cambio:	**0.0048381137 s**
Media de tiempo de respuesta después del cambio:	**0.004119396 s**



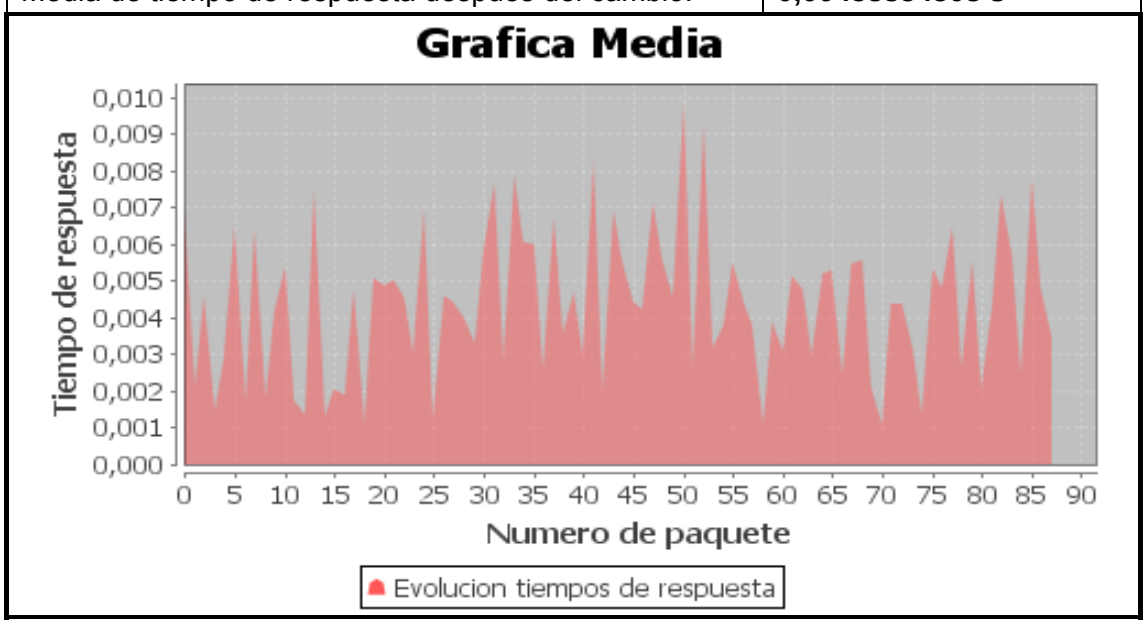
PRUEBA 1.2.1.2.D	
Número total de Request:	**100**
Número total de Reply:	**88**
Número de peticiones perdidas:	**12**
El cambio de red lo efectuó en el paquete:	**40**
Media de tiempo de respuesta antes del cambio:	**0.006487249 s**
Media de tiempo de respuesta después del cambio:	**0.003906647 s**



PRUEBA 1.2.1.2.E	
Número total de Request:	**100**
Número total de Reply:	**88**
Número de peticiones perdidas:	**12**
El cambio de red lo efectuó en el paquete:	**40**
Media de tiempo de respuesta antes del cambio:	**0.0028769833 s**
Media de tiempo de respuesta después del cambio:	**0.00562191 s**



MEDIA DE LAS PRUEBA 1.2.1.2: PING 1000 bytes HIP ESTRUCTURA VIRTUAL	
El Número de peticiones perdidas:	**12**
Media de tiempo de respuesta antes del cambio:	**0,00415096826 s**
Media de tiempo de respuesta después del cambio:	**0,00453354508 s**



Conclusiones

El número de paquetes perdidos es de 12, por lo que, al igual que antes, el tiempo que tarda en retomar la conexión una vez iniciado el cambio es de 12 segundos, por lo tanto, el aumento el tamaño de los paquetes no ha influido en este aspecto.

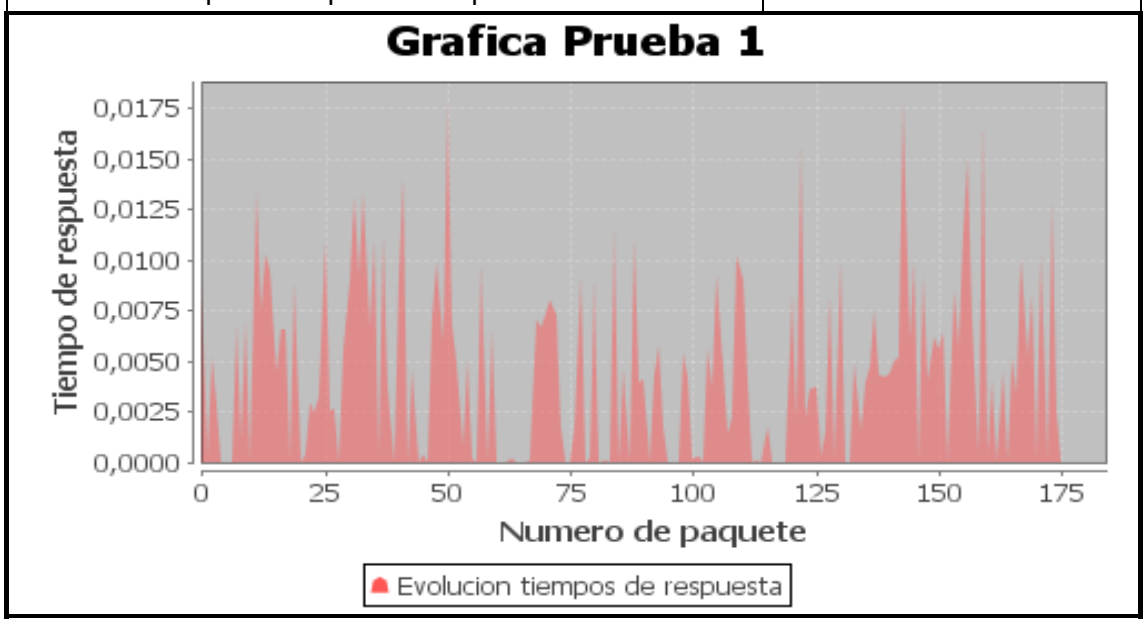
Se observa también que el tiempo en la red local sigue siendo solo ligeramente inferior al tiempo en la red a través de HIP, lo que nos indica que éste protocolo se comporta muy bien con el tratamiento de paquetes de tamaño grande. Los tiempos, tanto en local como a través de HIP son muy similares en esta prueba y en la anterior, lo que indica que no hay variación entre el comportamiento de las redes con un tamaño o con otro.

1.2.1.3 Envío de paquetes ping cada 0.5 segundos durante el cambio de red

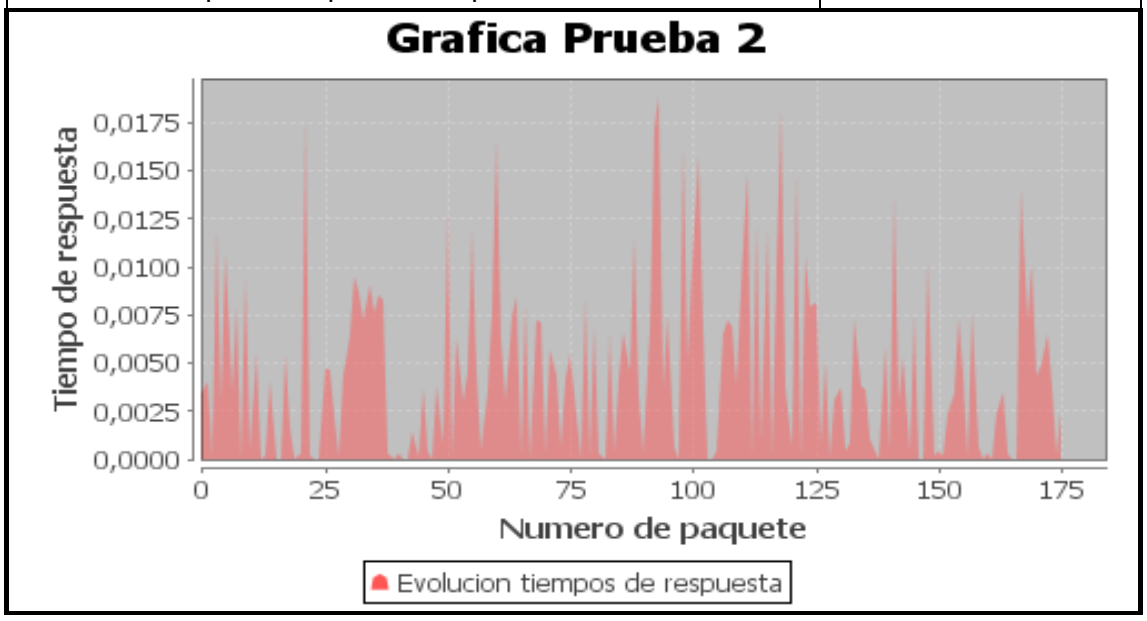
Una vez estudiado el comportamiento con paquetes de un tamaño mayor, el cometido de la siguiente prueba será estudiar el comportamiento de la tecnología HIP con el envío de paquetes con una frecuencia mayor. En este caso, vamos a mandar un paquete cada 0.5 segundos, para ver cómo se comporta HIP.

Como en los casos anteriores, para tener una muestra con el mismo tiempo, ahora el número de paquetes sobre el que se hace la prueba es de 200 (el doble que antes, ya que también se envían al doble de velocidad).

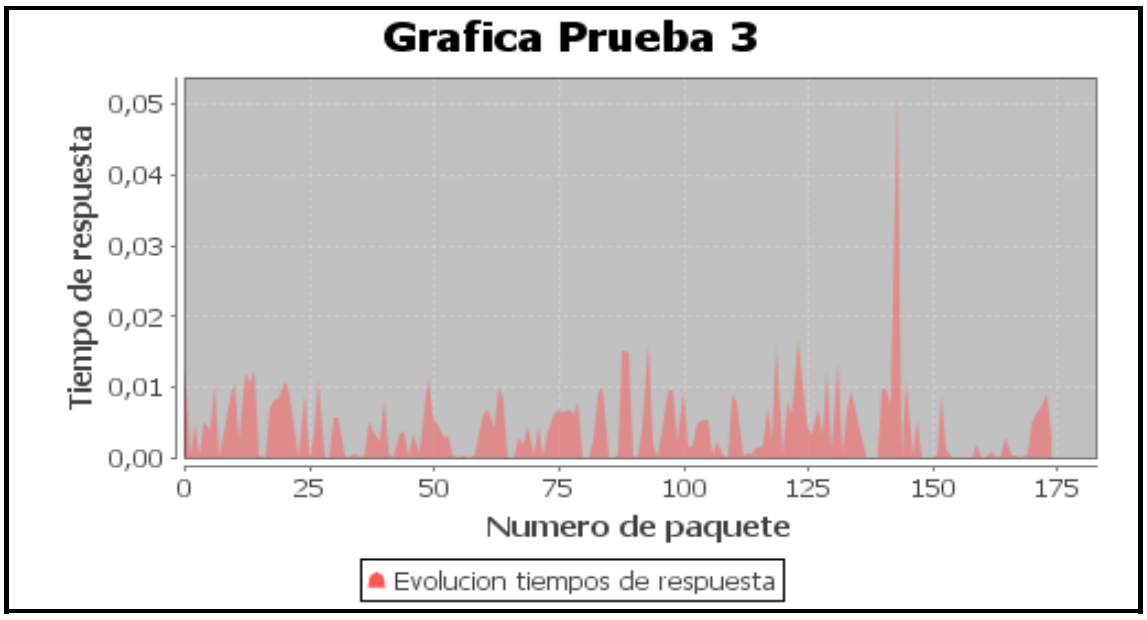
PRUEBA 1.2.1.3.A	
Número total de Request:	**200**
Número total de Reply:	**176**
Número de peticiones perdidas:	**24**
El cambio de red lo efectuó en el paquete:	**80**
Media de tiempo de respuesta antes del cambio:	**0.0044764006 s**
Media de tiempo de respuesta después del cambio:	**0.004324158 s**



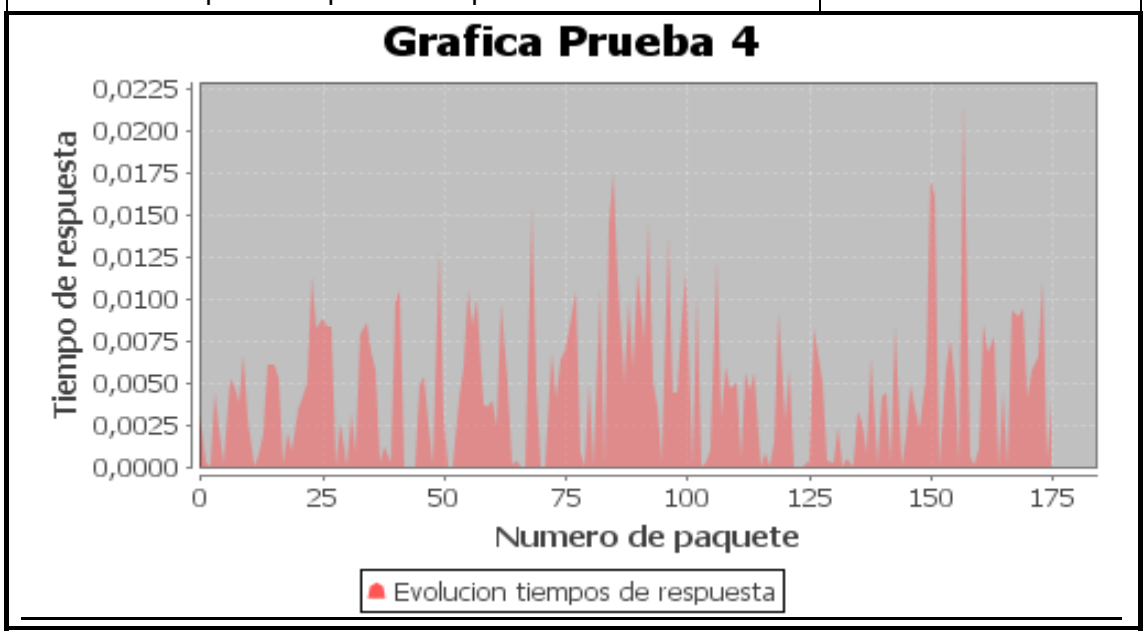
PRUEBA 1.2.1.3.B	
Número total de Request:	**200**
Número total de Reply:	**176**
Número de peticiones perdidas:	**24**
El cambio de red lo efectuó en el paquete:	**80**
Media de tiempo de respuesta antes del cambio:	**0.004126484 s**
Media de tiempo de respuesta después del cambio:	**0.004814267 s**



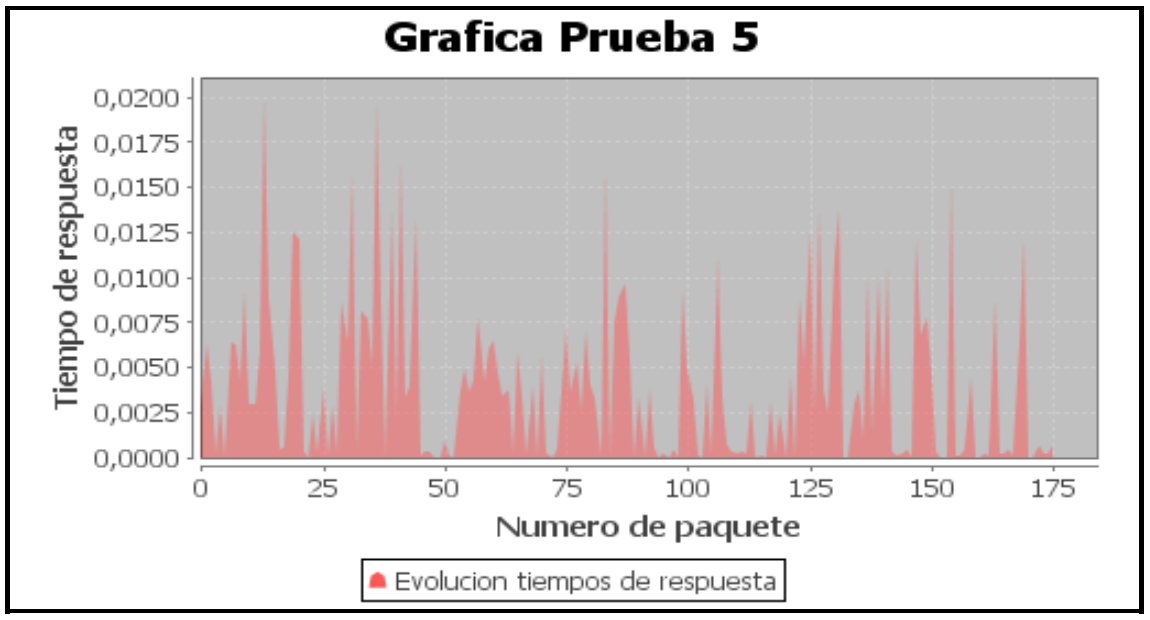
PRUEBA 1.2.1.3.C	
Número total de Request:	**200**
Número total de Reply:	**175**
Número de peticiones perdidas:	**25**
El cambio de red lo efectuó en el paquete:	**80**
Media de tiempo de respuesta antes del cambio:	**0.0041791424 s**
Media de tiempo de respuesta después del cambio:	**0.004416616 s**



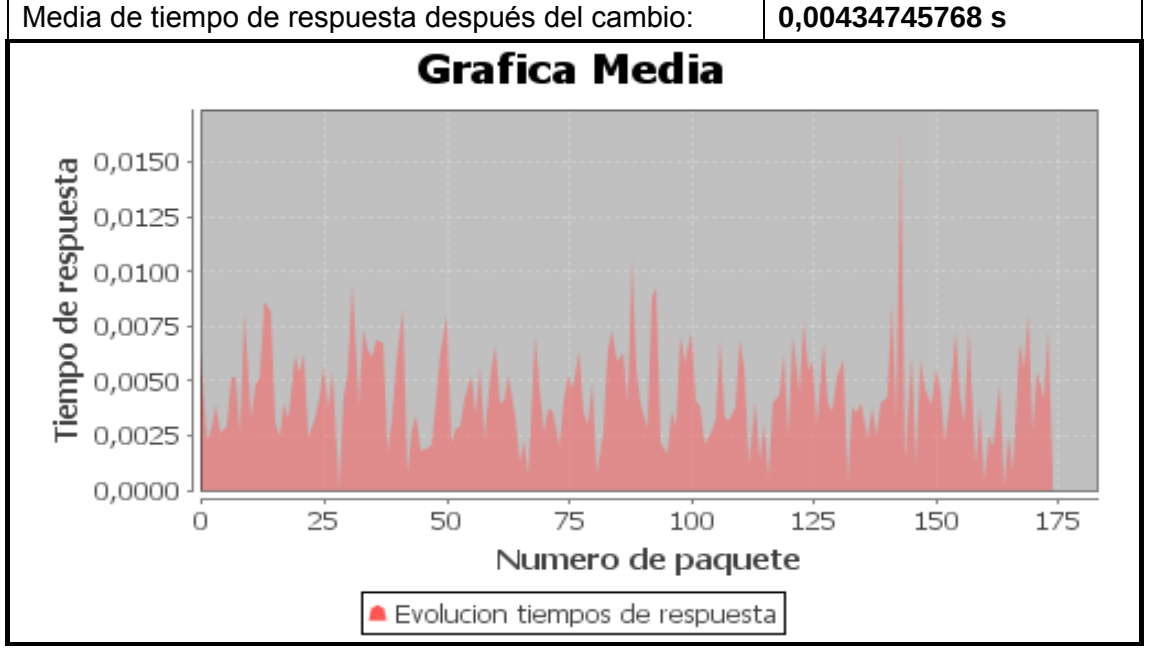
PRUEBA 1.2.1.3.D	
Número total de Request:	**200**
Número total de Reply:	**176**
Número de peticiones perdidas:	**24**
El cambio de red lo efectuó en el paquete:	**80**
Media de tiempo de respuesta antes del cambio:	**0.0041135834 s**
Media de tiempo de respuesta después del cambio:	**0.0049393573 s**



PRUEBA 1.2.1.3.E	
Número total de Request:	**200**
Número total de Reply:	**176**
Número de peticiones perdidas:	**24**
El cambio de red lo efectuó en el paquete:	**80**
Media de tiempo de respuesta antes del cambio:	**0.004540845 s**
Media de tiempo de respuesta después del cambio:	**0.0032428901 s**



MEDIA DE LAS PRUEBA 1.2.1.3: PING 0.5 seg HIP ESTRUCTURA VIRTUAL	
El Número de peticiones perdidas:	**24.2**
Media de tiempo de respuesta antes del cambio:	**0,00445244586 s**
Media de tiempo de respuesta después del cambio:	**0,00434745768 s**



Conclusiones

El número de paquetes perdidos de media es 24.2, y teniendo en cuenta que los paquetes se enviaban cada 0.5 segundos, se obtiene un tiempo de espera de 12.1, muy similar al obtenido en las pruebas anteriores, lo que ajusta mucho el tiempo que tarda en cambiar de red e inicializar los servicios.

Se puede apreciar que los paquetes siguen teniendo unos valores poco homogéneos, ya que en la gráfica se puede ver como hay una gran cantidad de picos. Este motivo ya ha sido explicado antes, y es que se obtienen valores muy diversos para paquetes del mismo tipo, siendo unos valores cercanos a 0, y estando otros por encima de la media.

Respecto a los tiempos, se puede observar un dato curioso, y es que el tiempo medio en la red a través de HIP es menor que el tiempo a través de la red local. Esto es debido a la

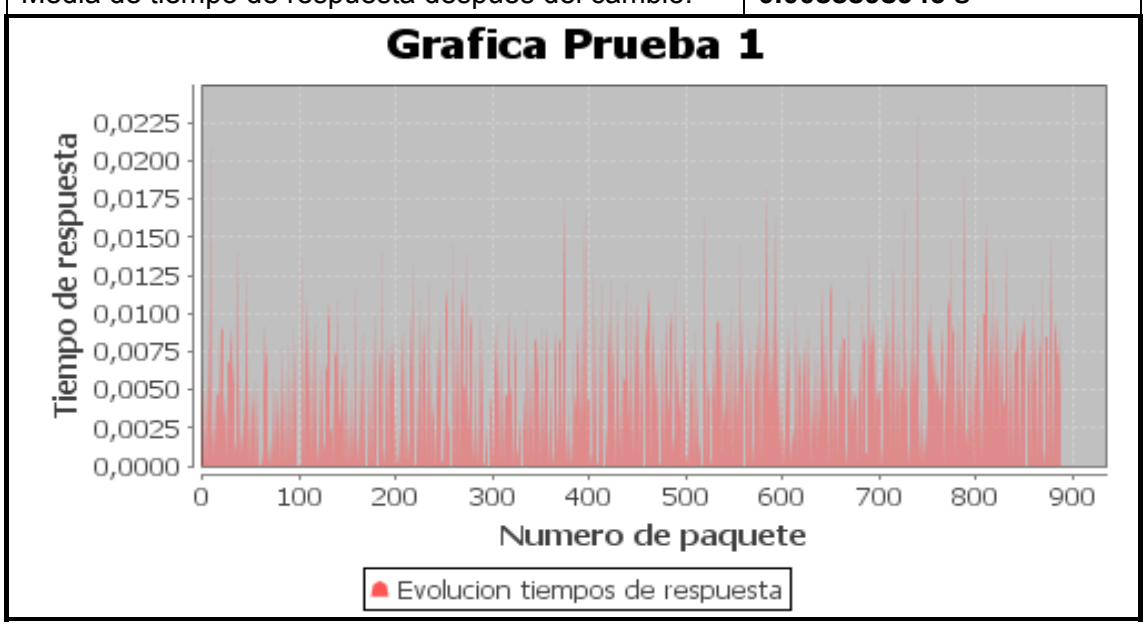
gran cantidad de picos que tenemos en la red local, que hace que suba mucho su tiempo medio, y que al final se obtenga este dato curioso.

Al igual que ocurría antes, los tiempos tanto en local como a través de HIP son muy similares a la prueba de ping básica, lo que indica un buen comportamiento con una frecuencia de este nivel.

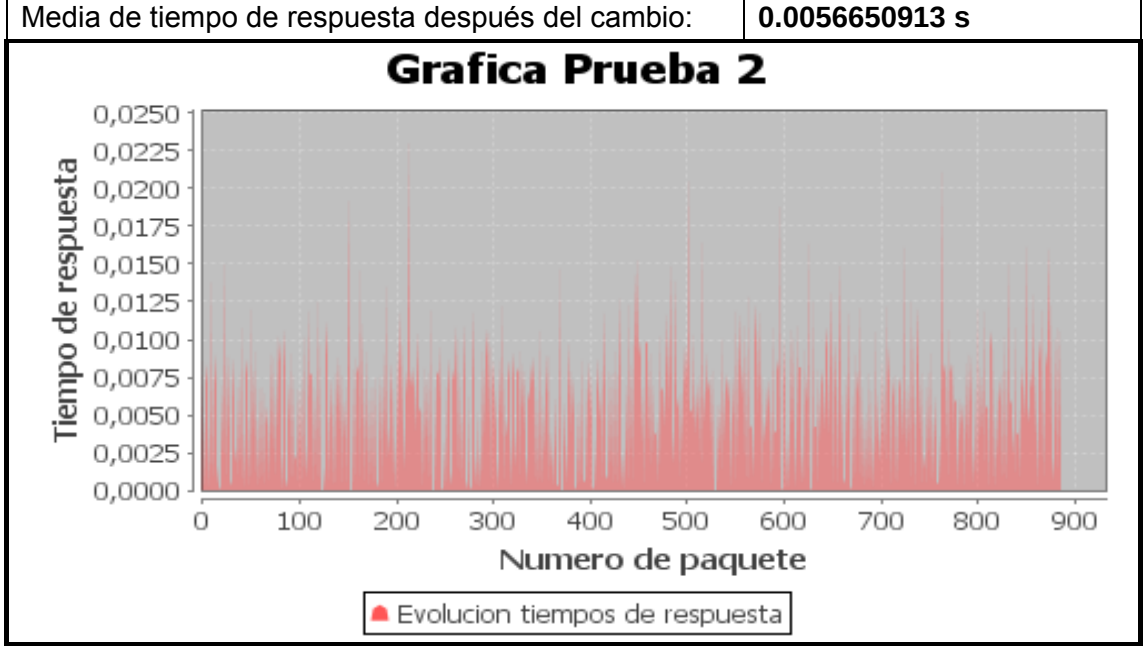
1.2.1.4 Envío de paquetes ping cada 0.1 segundos durante el cambio de red

La última prueba a realizar sobre HIP en estructuras virtuales trata del envío de paquetes de 64 bytes a una frecuencia de envío de 10 paquetes por segundo. Para ello, hay que dar al usuario permiso de superusuario para poder utilizar una frecuencia tan alta.

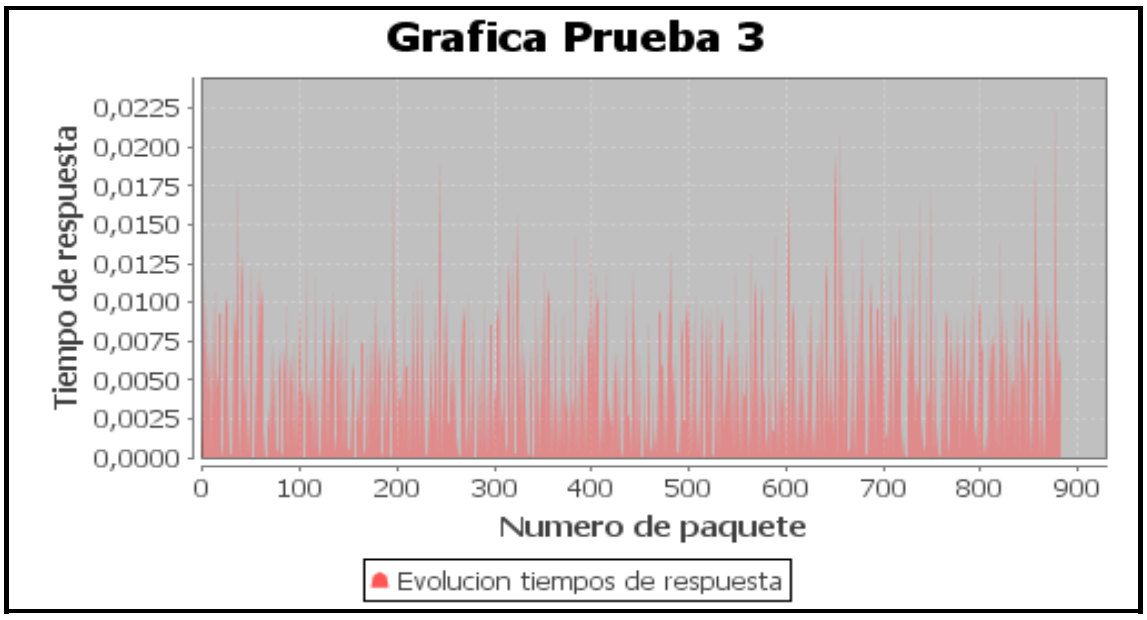
PRUEBA 1.2.1.4.A	
Número total de Request:	**999**
Número total de Reply:	**890**
Número de peticiones perdidas:	**109**
El cambio de red lo efectuó en el paquete:	**401**
Media de tiempo de respuesta antes del cambio:	**0.0042831074 s**
Media de tiempo de respuesta después del cambio:	**0.0055593946 s**



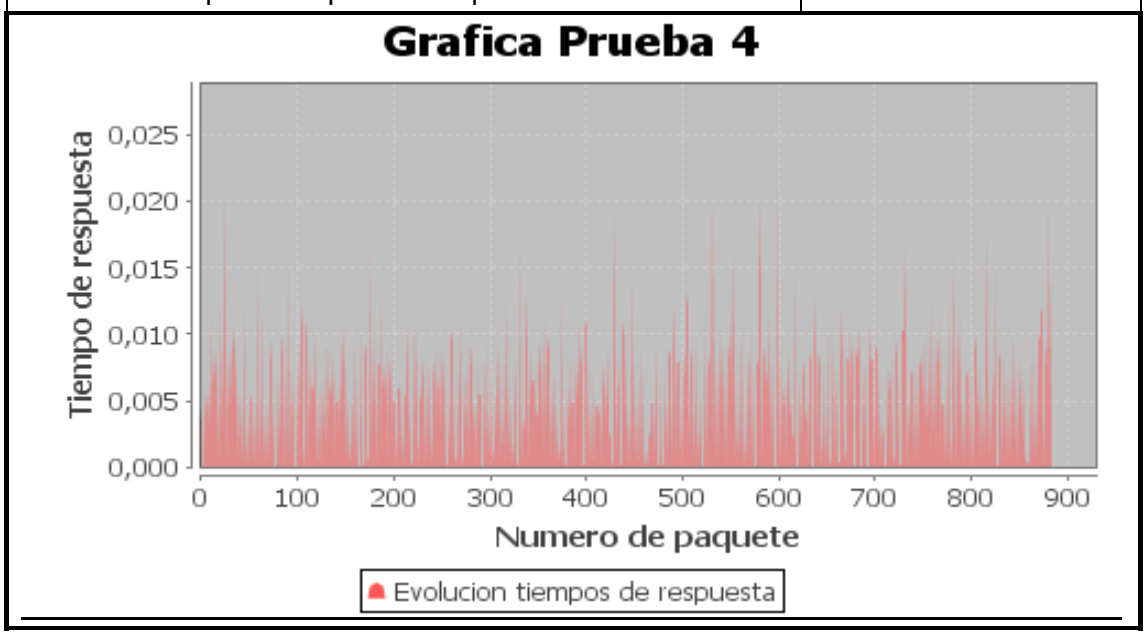
PRUEBA 1.2.1.4.B	
Número total de Request:	**1000**
Número total de Reply:	**888**
Número de peticiones perdidas:	**112**
El cambio de red lo efectuó en el paquete:	**400**
Media de tiempo de respuesta antes del cambio:	**0.004770835 s**
Media de tiempo de respuesta después del cambio:	**0.0056650913 s**



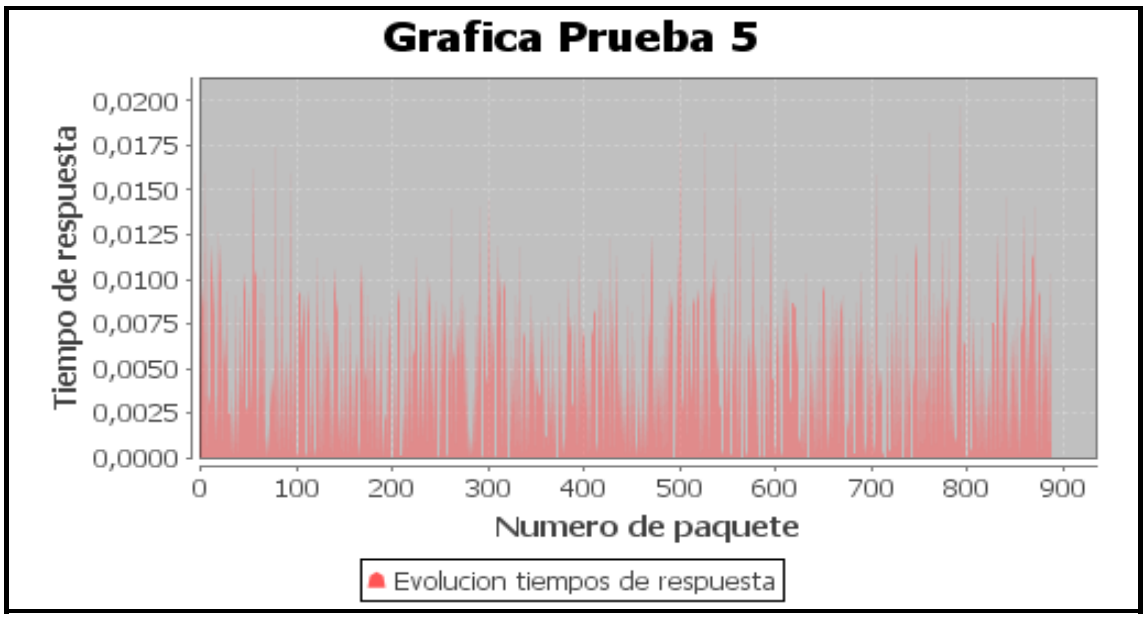
PRUEBA 1.2.1.4.C	
Número total de Request:	**1000**
Número total de Reply:	**885**
Número de peticiones perdidas:	**115**
El cambio de red lo efectuó en el paquete:	**401**
Media de tiempo de respuesta antes del cambio:	**0.0046321694 s**
Media de tiempo de respuesta después del cambio:	**0.00495686 s**



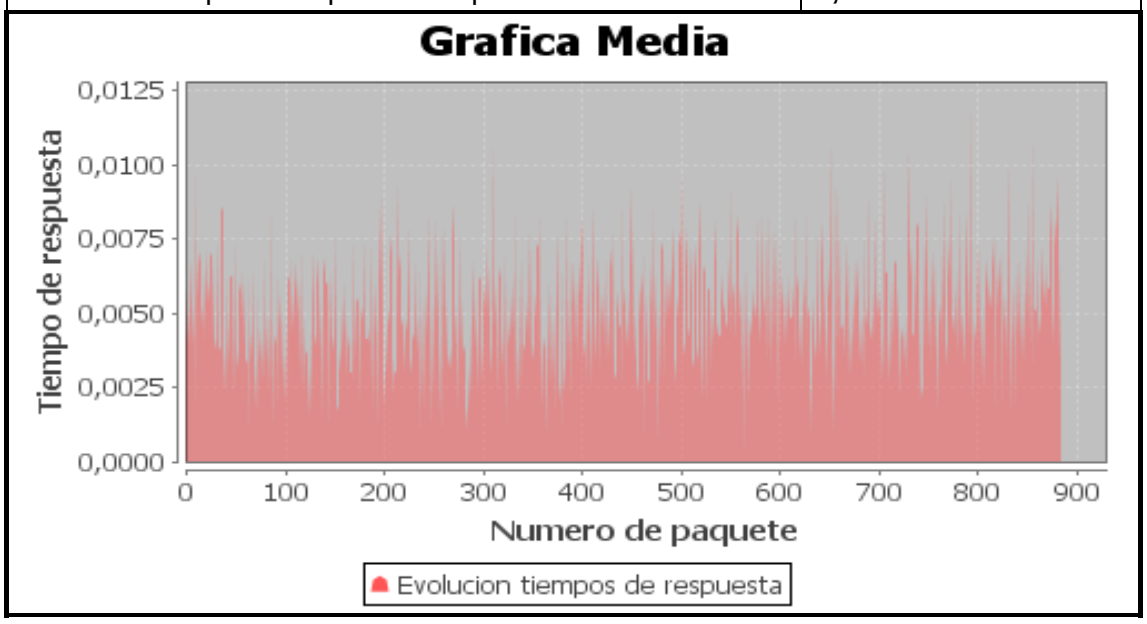
PRUEBA 1.2.1.4.D	
Número total de Request:	**1000**
Número total de Reply:	**886**
Número de peticiones perdidas:	**114**
El cambio de red lo efectuó en el paquete:	**400**
Media de tiempo de respuesta antes del cambio:	**0.0046903538 s**
Media de tiempo de respuesta después del cambio:	**0.0051242374 s**



PRUEBA 1.2.1.4.E	
Número total de Request:	**999**
Número total de Reply:	**890**
Número de peticiones perdidas:	**109**
El cambio de red lo efectuó en el paquete:	**400**
Media de tiempo de respuesta antes del cambio:	**0.0047442475 s**
Media de tiempo de respuesta después del cambio:	**0.0048467675 s**



MEDIA DE LAS PRUEBA 1.2.1.4: PING 0.1 seg HIP ESTRUCTURA VIRTUAL	
El Número de peticiones perdidas:	**133.6**
Media de tiempo de respuesta antes del cambio:	**0,00462414262 s**
Media de tiempo de respuesta después del cambio:	**0,00523047016 s**



Conclusiones

El número de paquetes perdidos de media es de 133,6, lo cual, sabiendo que la frecuencia de envío es de 1 paquete cada 0.1 segundos, indica que el tiempo de espera después del cambio de red es de unos 13,3 segundos.

Al igual que en todas las pruebas con HIP, se observa la gran variedad de picos que hay en la gráfica. Como ya se ha explicado en las anteriores pruebas, HIP es mucho más heterogéneo en los resultados obtenidos en las máquinas virtuales que openvpn, aunque obtiene unos resultados mucho mejores.

Respecto a los tiempos, ninguno de ellos ha tenido diferencias apreciables con la prueba básica del ping, y siguen estando muy parecidos, siendo el tiempo a través del HIP ligeramente superior al tiempo en la red local.

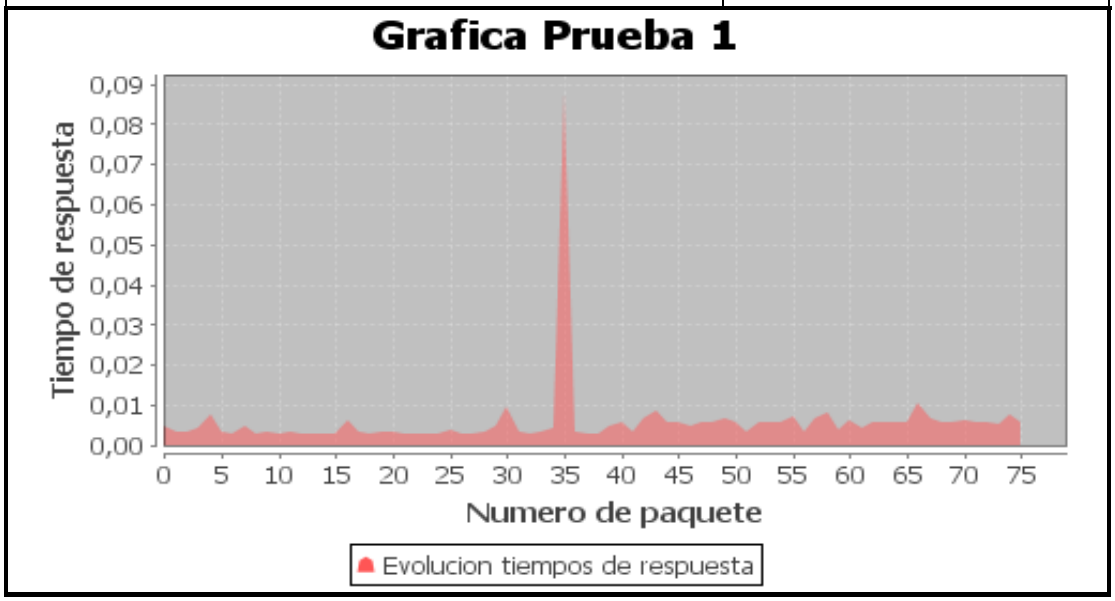
1.2.2- Real

1.2.2.1 Envío de paquetes ping de básicos durante el cambio de red

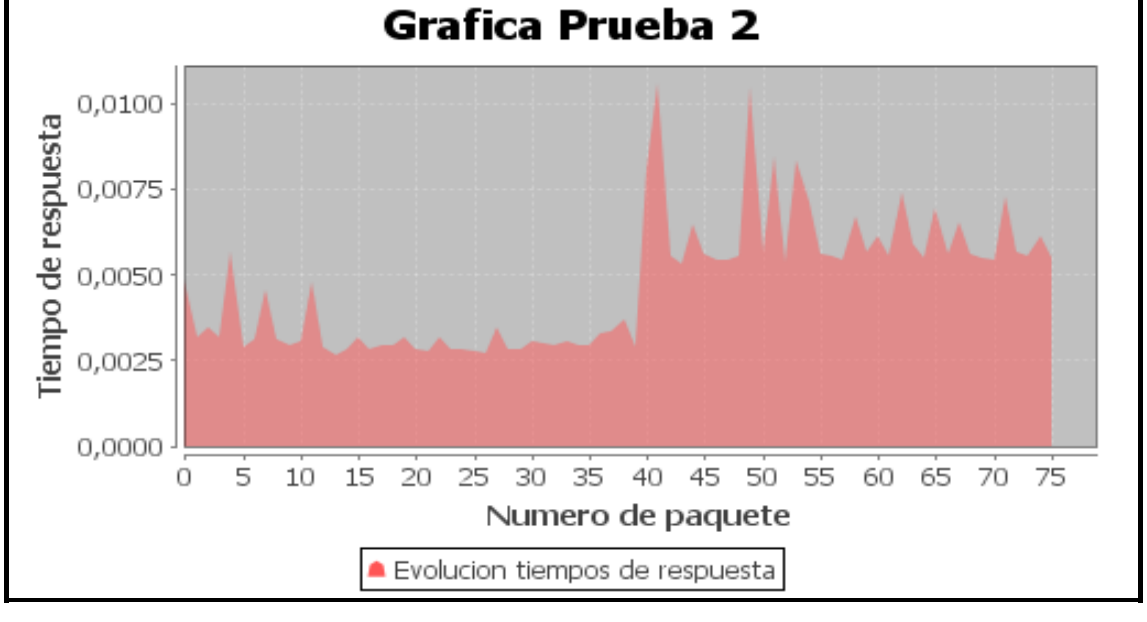
Para finalizar el estudio de la tecnología HIP con la herramienta ping, se realizaron las pruebas sobre la estructura real, con las máquinas con HIP instalado y la estructura mencionada anteriormente.

La primera prueba a realizar es el ping básico, es decir, el envío de paquetes de 64 bytes cada segundo.

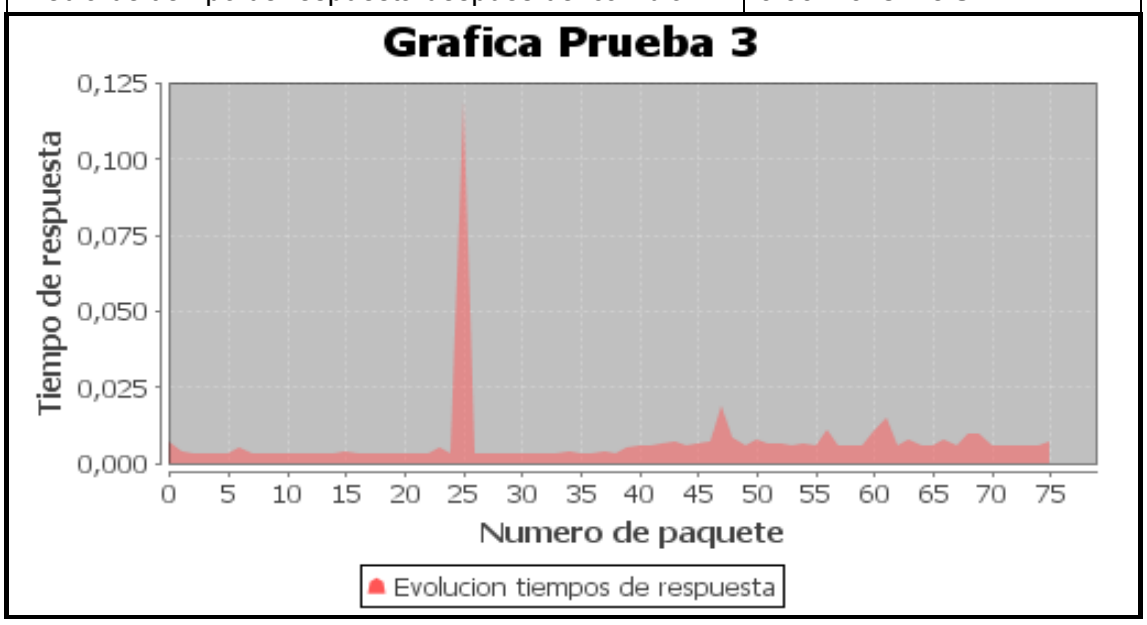
PRUEBA 1.2.2.1.A	
Número total de Request:	**99**
Número total de Reply:	**76**
Número de peticiones perdidas:	**23**
El cambio de red lo efectuó en el paquete:	**40**
Media de tiempo de respuesta antes del cambio:	**0.0056539206 s**
Media de tiempo de respuesta después del cambio:	**0.0057572257 s**



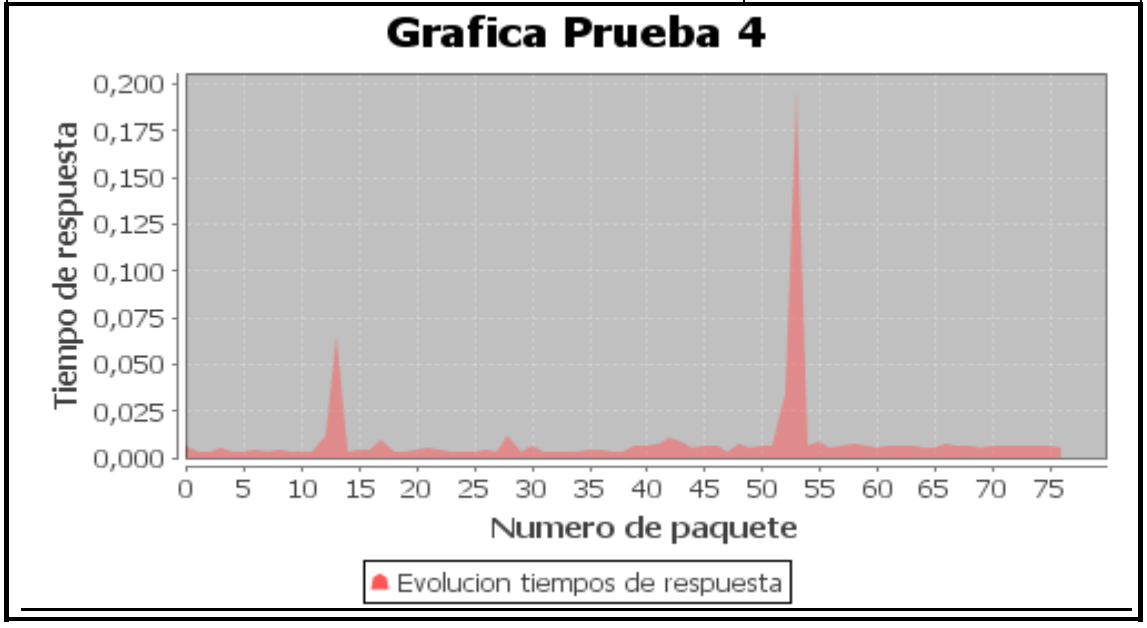
PRUEBA 1.2.2.1.B	
Número total de Request:	**100**
Número total de Reply:	**76**
Número de peticiones perdidas:	**24**
El cambio de red lo efectuó en el paquete:	**40**
Media de tiempo de respuesta antes del cambio:	**0.0032064163 s**
Media de tiempo de respuesta después del cambio:	**0.006344689 s**



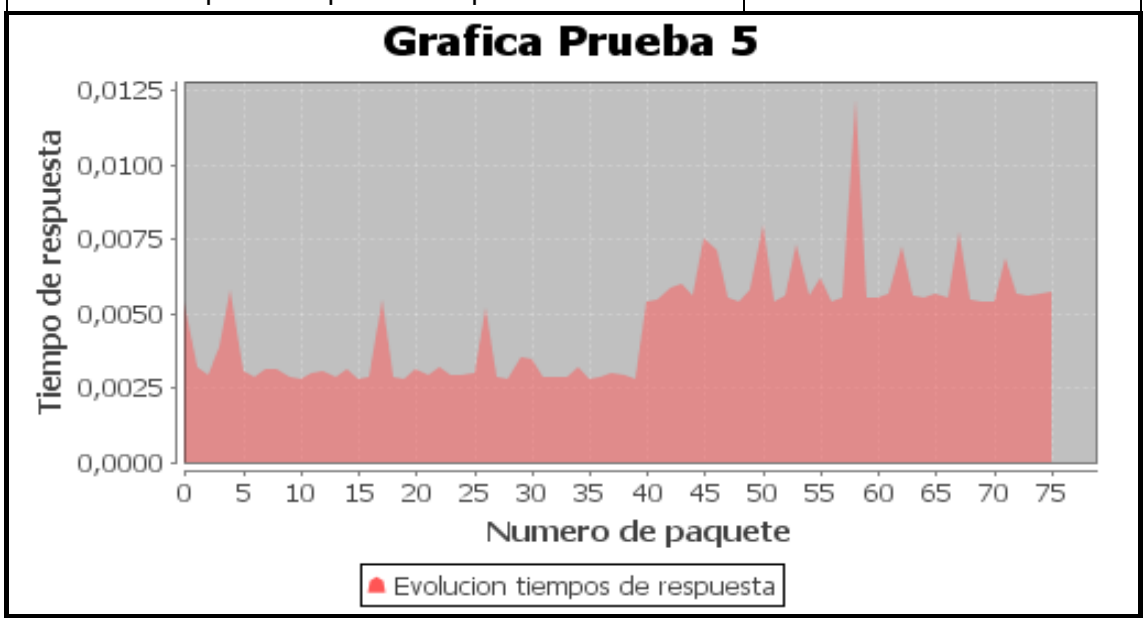
PRUEBA 1.2.2.1.C	
Número total de Request:	**100**
Número total de Reply:	**76**
Número de peticiones perdidas:	**24**
El cambio de red lo efectuó en el paquete:	**40**
Media de tiempo de respuesta antes del cambio:	**0.006202585 s**
Media de tiempo de respuesta después del cambio:	**0.0071025426 s**



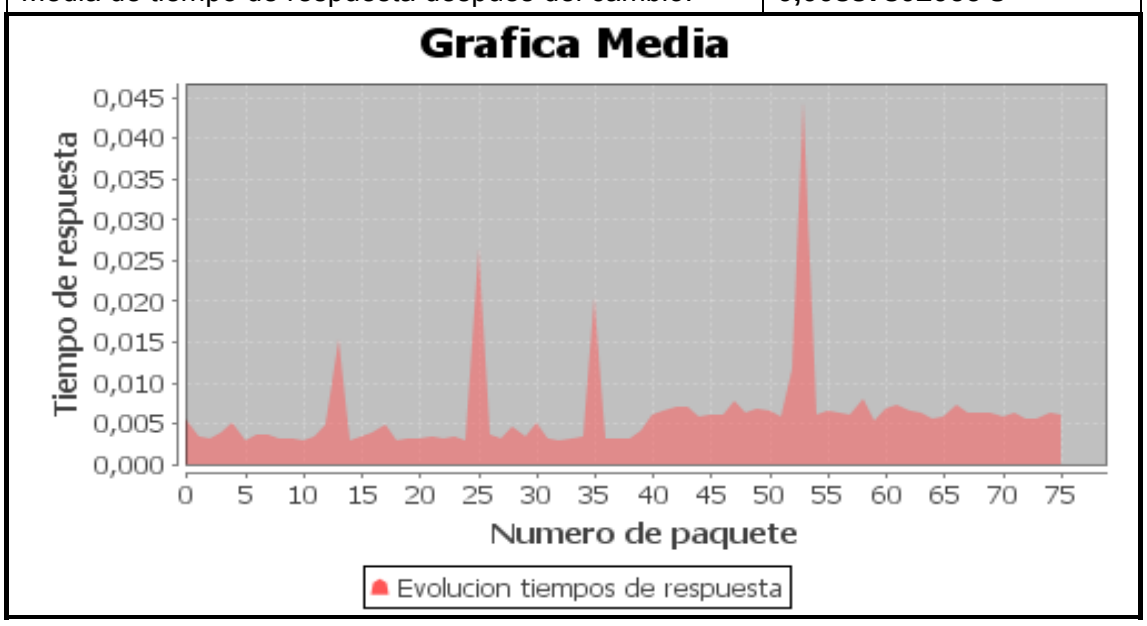
PRUEBA 1.2.2.1.D	
Número total de Request:	**100**
Número total de Reply:	**77**
Número de peticiones perdidas:	**23**
El cambio de red lo efectuó en el paquete:	**40**
Media de tiempo de respuesta antes del cambio:	**0.005579029 s**
Media de tiempo de respuesta después del cambio:	**0.011907217 s**



PRUEBA 1.2.2.1.E	
Número total de Request:	**100**
Número total de Reply:	**76**
Número de peticiones perdidas:	**24**
El cambio de red lo efectuó en el paquete:	**40**
Media de tiempo de respuesta antes del cambio:	**0.0032293722 s**
Media de tiempo de respuesta después del cambio:	**0.006113476 s**



MEDIA DE LAS PRUEBA 1.2.1.1: PING NORMAL HIP ESTRUCTURA REAL	
El Número de peticiones perdidas:	**23.6**
Media de tiempo de respuesta antes del cambio:	**0,00477426462 s**
Media de tiempo de respuesta después del cambio:	**0,00887802066 s**



Conclusiones

Ahora, una vez que HIP se prueba sobre una estructura real, se puede observar que los resultados son mucho más homogéneos. Ya no hay tantos picos como antes. Ya no hay ningún valor para los paquetes que valga casi 0, y los picos no son tan habituales, pues en cada prueba solo hay como mucho un pico que sobresalga de manera notable sobre los demás, lo cual es algo normal.

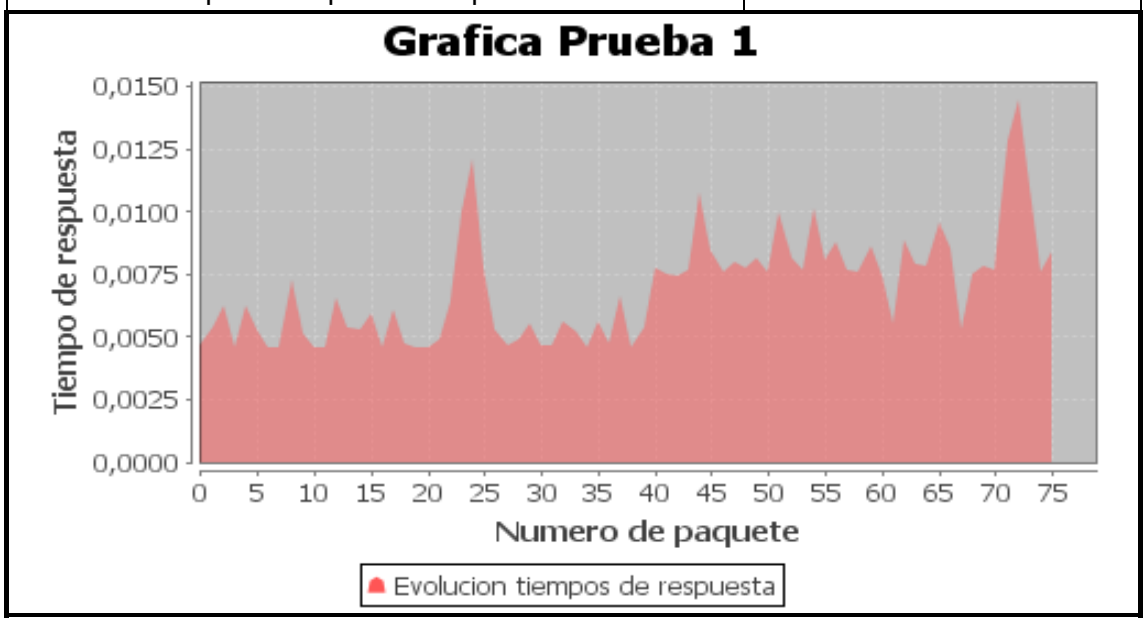
El número de paquetes perdidos es de 23,6 de media, lo que indica un tiempo de 23.6 segundos desde que se inicia el cambio de red hasta que se recupera la conexión. Este tiempo es debido a los tres factores que se explicaba en la misma prueba sobre OpenVPN, que son la desconexión de la primera red, la conexión a la segunda red (lo que más tardaba) y la reconexión, a través de HIP.

En cuanto a los tiempos en las distintas redes, se ve que el tiempo en la red local ahora sí que es significativamente menor que el tiempo en la red a través de HIP, cosa que en las máquinas virtuales no quedaba tan claro. En esta primera prueba, se puede apreciar que el tiempo en la red local es de 0.00477426462 segundos, mientras que en la red a través de HIP es de 0.00887802066 segundos, lo que es equivalente a un 85.9% más.

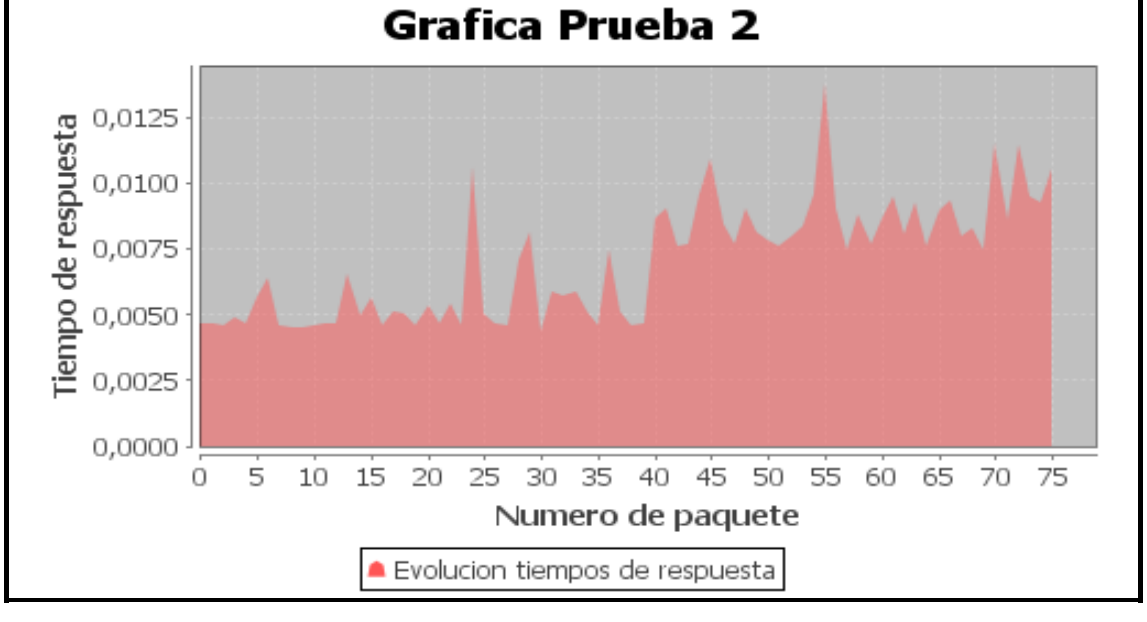
1.2.2.2 Envío de paquetes ping de 1000 bytes durante el cambio de red

Después del ping normal, y como se hizo con todas las pruebas de ping anteriores, se quiso estudiar el comportamiento de HIP en la estructura real con el envío de paquetes de un tamaño mayor. Así, en esta prueba el tamaño de los paquetes fue de 1000 bytes.

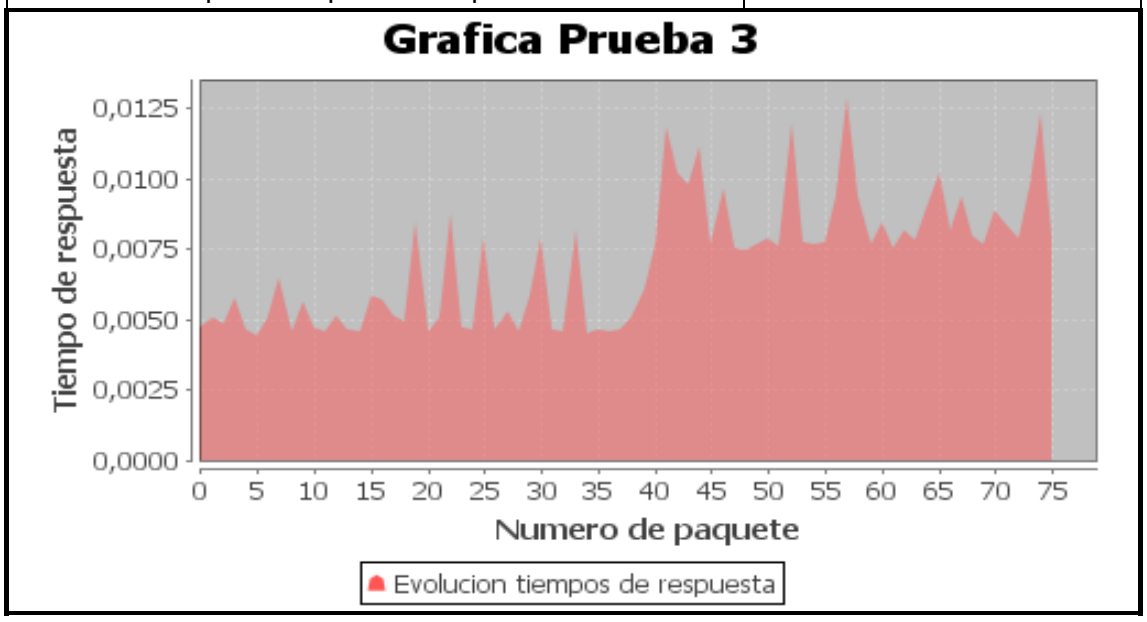
PRUEBA 1.2.2.2.A	
Número total de Request:	**100**
Número total de Reply:	**76**
Número de peticiones perdidas:	**24**
El cambio de red lo efectuó en el paquete:	**40**
Media de tiempo de respuesta antes del cambio:	**0.0055856137 s**
Media de tiempo de respuesta después del cambio:	**0.008416493 s**



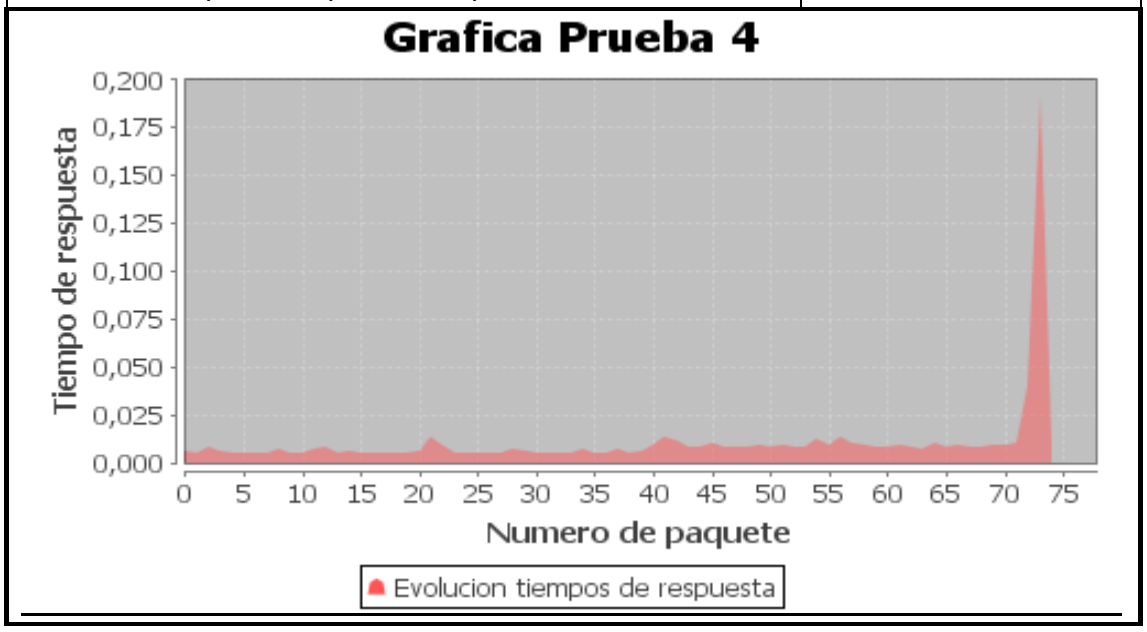
PRUEBA 1.2.2.2.B	
Número total de Request:	**100**
Número total de Reply:	**76**
Número de peticiones perdidas:	**24**
El cambio de red lo efectuó en el paquete:	**40**
Media de tiempo de respuesta antes del cambio:	**0.005316197 s**
Media de tiempo de respuesta después del cambio:	**0.008899053 s**



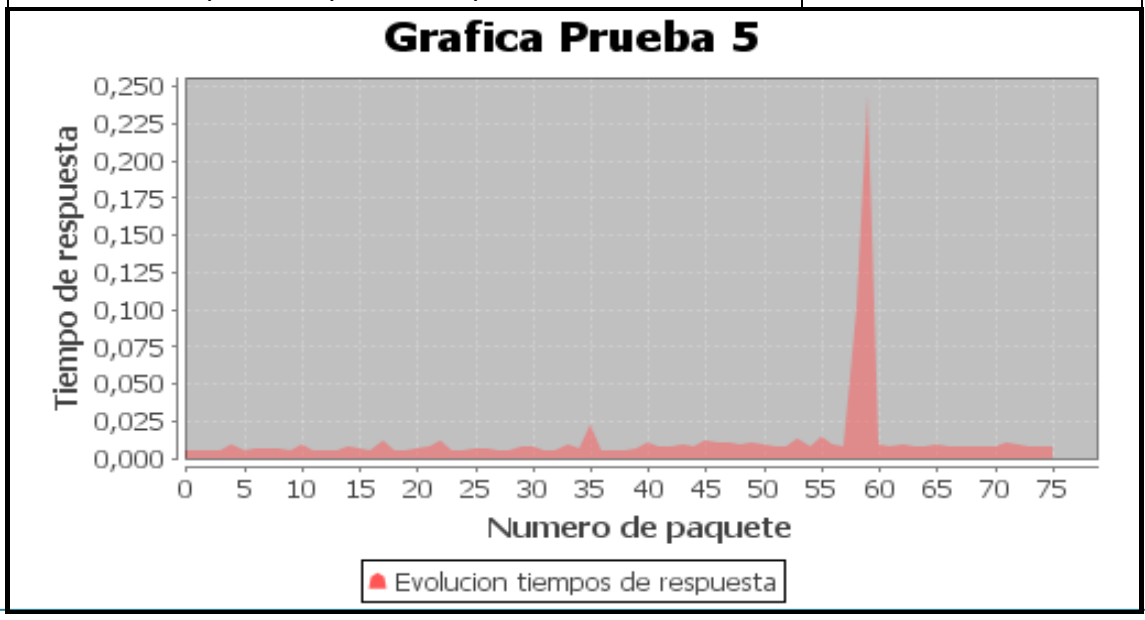
PRUEBA 1.2.2.2.C	
Número total de Request:	**100**
Número total de Reply:	**76**
Número de peticiones perdidas:	**24**
El cambio de red lo efectuó en el paquete:	**40**
Media de tiempo de respuesta antes del cambio:	**0.0053721475 s**
Media de tiempo de respuesta después del cambio:	**0.008860482 s**



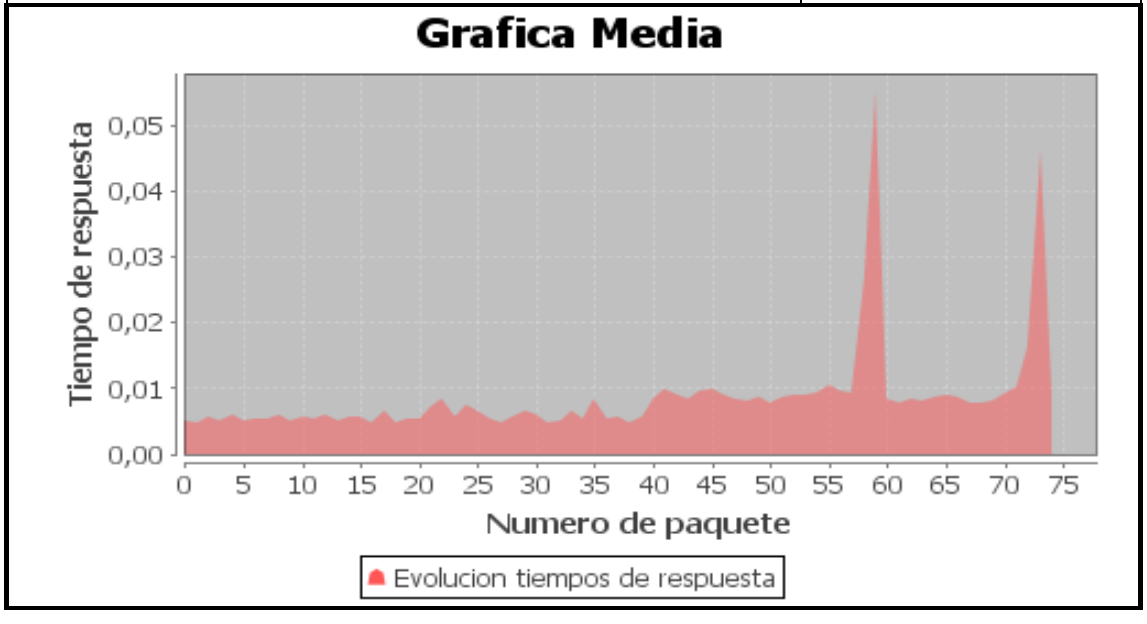
PRUEBA 1.2.2.2.D	
Número total de Request:	**100**
Número total de Reply:	**75**
Número de peticiones perdidas:	**25**
El cambio de red lo efectuó en el paquete:	**40**
Media de tiempo de respuesta antes del cambio:	**0.0056181094 s**
Media de tiempo de respuesta después del cambio:	**0.015131052 s**



PRUEBA 1.2.2.2.E	
Número total de Request:	**100**
Número total de Reply:	**76**
Número de peticiones perdidas:	**24**
El cambio de red lo efectuó en el paquete:	**40**
Media de tiempo de respuesta antes del cambio:	**0.0063751875 s**
Media de tiempo de respuesta después del cambio:	**0.017799802 s**



MEDIA DE LAS PRUEBA 1.2.1.2: PING 1000 bytes HIP ESTRUCTURA REAL	
El Número de peticiones perdidas:	**24,2**
Media de tiempo de respuesta antes del cambio:	**0,00565345102 s**
Media de tiempo de respuesta después del cambio:	**0,0118196718 s**



Conclusiones

El número de paquetes perdidos en esta ocasión ha sido de 24.2 de media, lo que indica un tiempo de espera de reconexión de 24.2 segundos, muy similar al obtenido en el ping básico, por lo que el tamaño de los paquetes, obviamente, no influye en el tiempo de reconexión.

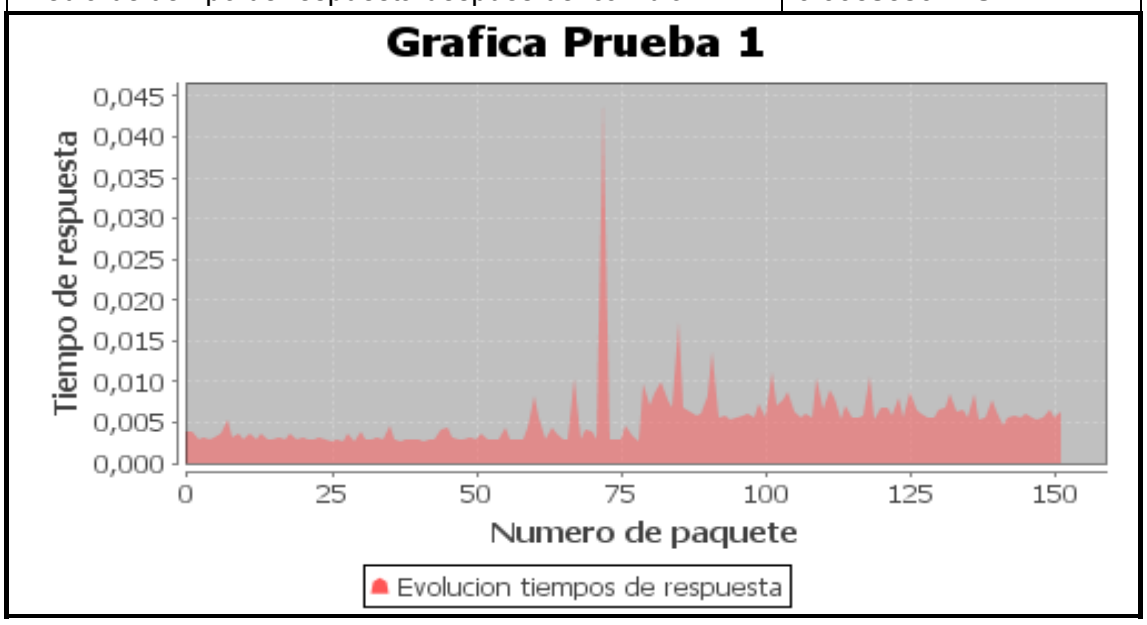
Se sigue observando una uniformidad y una homogeneidad en los datos y en la gráfica mayor de la que se había obtenido cuando se realizaban las pruebas en las máquinas virtuales. Excepto un par de picos aislados (dos picos grandes entre las 5 pruebas), el resto de valores son muy similares. Estos dos picos hacen que la media en el tiempo de respuesta esté un poco por encima de lo que realmente es el comportamiento de HIP para esta prueba.

En cuanto a los tiempos, el tiempo en red local ha aumentado de 0,00477426462 segundos a 0,00565345102 segundos, lo que se corresponde con una subida de un 18% , mientras que el tiempo en la red a través de HIP ha subido de 0,00887802066 s a 0,0118196718 s, lo que supone una subida de un 33%. Como ya se ha mencionado en el párrafo anterior, esta subida está un poco adulterada por los dos picos obtenidos; no es tan grande, pero el tiempo sí que aumenta.

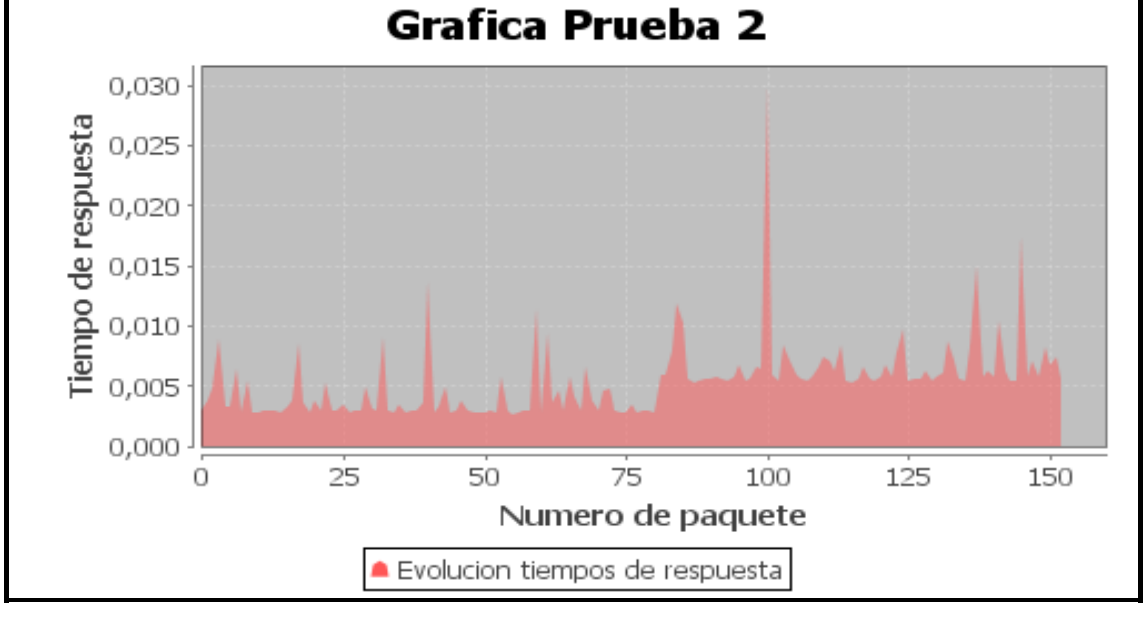
1.2.2.3 Envío de paquetes ping cada 0.5 segundos durante el cambio de red

Después de la prueba para poder comprobar el comportamiento de HIP en la estructura real con un tamaño de paquete grande, la siguiente prueba quiere comprobar el comportamiento cuando se aumenta la frecuencia de envío de paquetes, y llega a enviarse un paquete cada 0.5 segundos, lo que aumenta en el doble la frecuencia de envío de las pruebas anteriores.

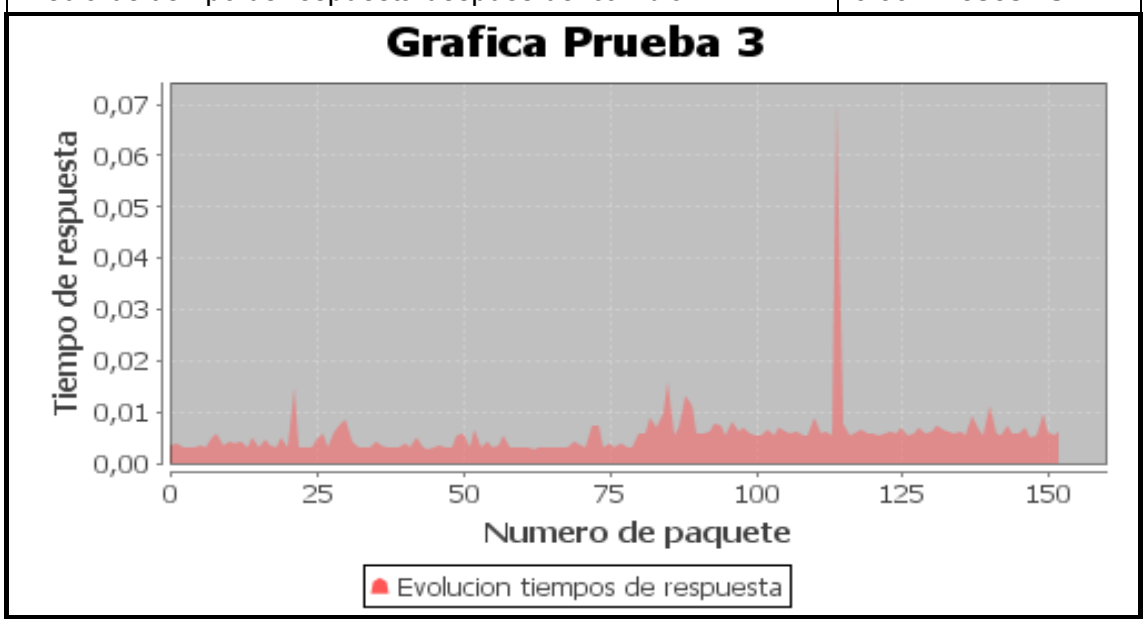
PRUEBA 1.2.2.3.A	
Número total de Request:	**199**
Número total de Reply:	**152**
Número de peticiones perdidas:	**47**
El cambio de red lo efectuó en el paquete:	**79**
Media de tiempo de respuesta antes del cambio:	**0.003859748 s**
Media de tiempo de respuesta después del cambio:	**0.006865044 s**



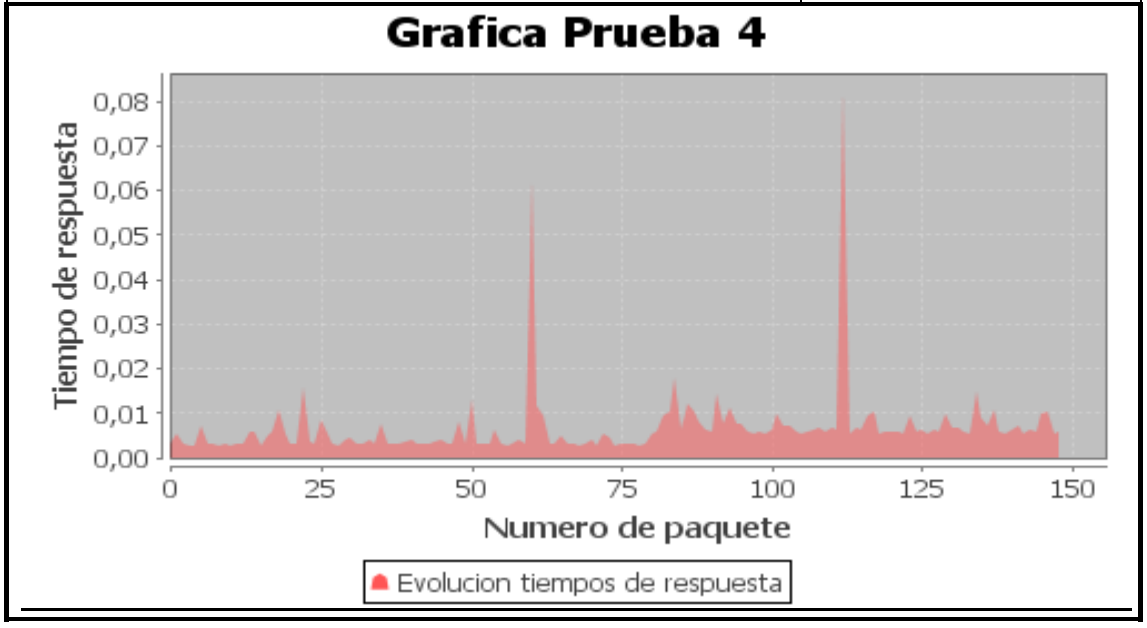
PRUEBA 1.2.2.3.B	
Número total de Request:	**200**
Número total de Reply:	**153**
Número de peticiones perdidas:	**47**
El cambio de red lo efectuó en el paquete:	**81**
Media de tiempo de respuesta antes del cambio:	**0.0038811723 s**
Media de tiempo de respuesta después del cambio:	**0.0070054797 s**



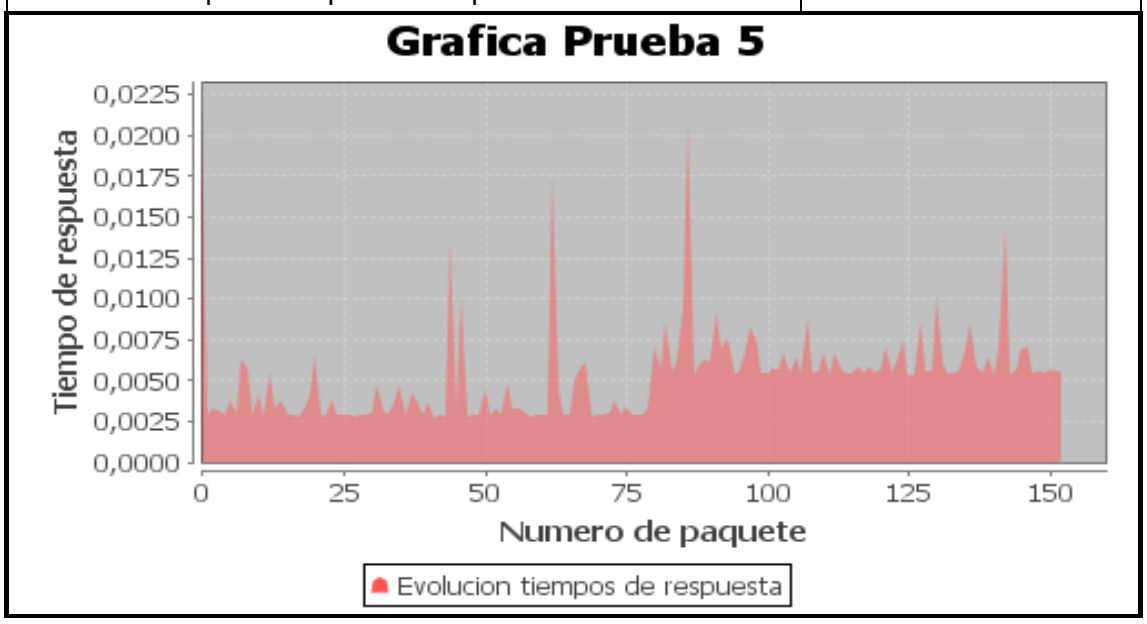
PRUEBA 1.2.2.3.C	
Número total de Request:	**200**
Número total de Reply:	**153**
Número de peticiones perdidas:	**47**
El cambio de red lo efectuó en el paquete:	**80**
Media de tiempo de respuesta antes del cambio:	**0.0038269027 s**
Media de tiempo de respuesta después del cambio:	**0.0074468637 s**



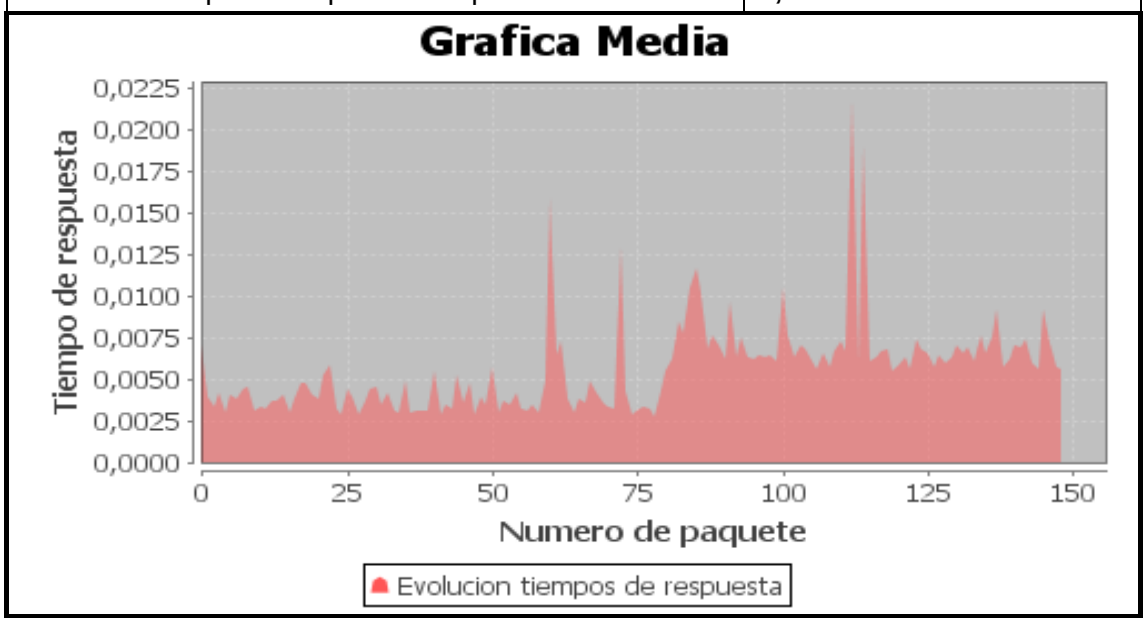
PRUEBA 1.2.2.3.D	
Número total de Request:	**200**
Número total de Reply:	**149**
Número de peticiones perdidas:	**51**
El cambio de red lo efectuó en el paquete:	**80**
Media de tiempo de respuesta antes del cambio:	**0.004814416 s**
Media de tiempo de respuesta después del cambio:	**0.008320242 s**



PRUEBA 1.2.2.3.E	
Número total de Request:	**200**
Número total de Reply:	**153**
Número de peticiones perdidas:	**47**
El cambio de red lo efectuó en el paquete:	**80**
Media de tiempo de respuesta antes del cambio:	**0.004036198 s**
Media de tiempo de respuesta después del cambio:	**0.00654393 s**



MEDIA DE LAS PRUEBA 1.2.1.3: PING 0.5 seg HIP ESTRUCTURA REAL	
El Número de peticiones perdidas:	**47,8**
Media de tiempo de respuesta antes del cambio:	**0,0040836874 s**
Media de tiempo de respuesta después del cambio:	**0,00723631188 s**



Conclusiones

En este caso, el número de paquetes perdidos ha sido 47.8 de media, y como la frecuencia era de 0.5, el tiempo que ha tardado la conexión en volver a estar activa ha sido de 23.9 segundos, un valor casi igual que el obtenido en las anteriores pruebas, por lo que se puede deducir que éste tampoco es un factor del que dependa la conexión en levantarse, sino que es totalmente independiente de la prueba que estemos realizando.

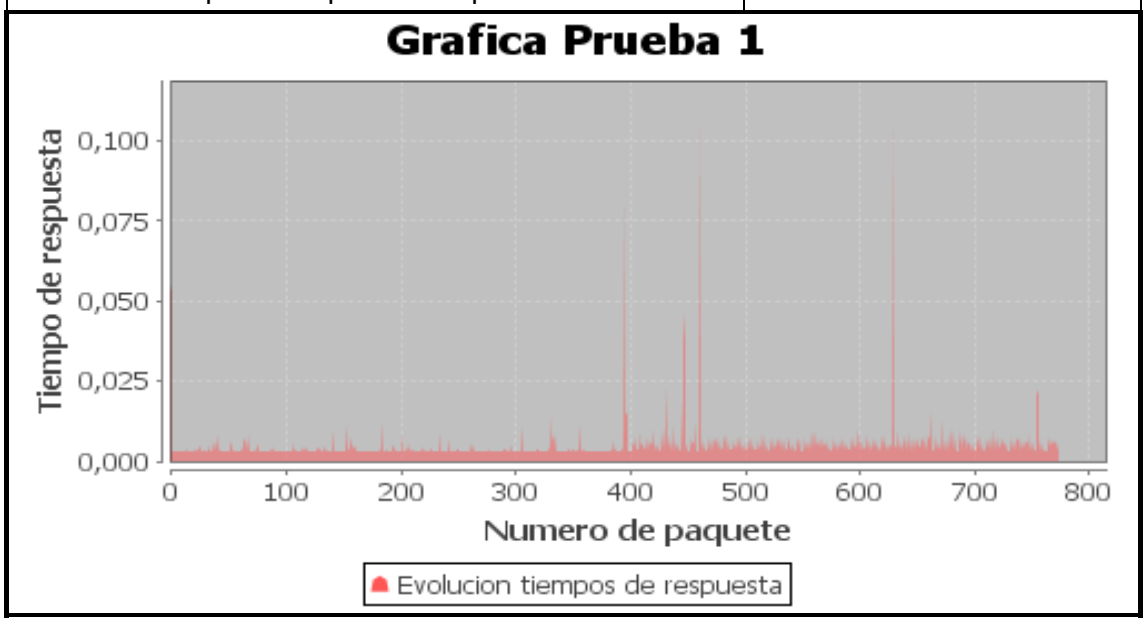
En cuanto a la homogeneidad, se consiguen resultados más regulares y normales que en la ejecución de las pruebas sobre máquinas virtuales, por lo que también se puede asegurar que ésta es una prueba con mayor significado, y no tan teórica como la otra, en la que no se tienen redes y máquinas físicas, sino virtuales, por lo que los resultados no son tan fiables.

Por último, respecto a los tiempos en ambas redes, vemos que son prácticamente iguales, e incluso un poco menores que en la prueba de pings normales. Esto también demuestra que HIP es un protocolo que se comporta bien (hablando de tiempo de respuesta de los ping) tanto con grandes tamaños de datos como con pequeños, como con datos llegando a una frecuencia superior. La relación entre los tiempos en red local y red a través de HIP sigue siendo alrededor de un 77% superior en la segunda, similar a los resultados obtenidos hasta ahora.

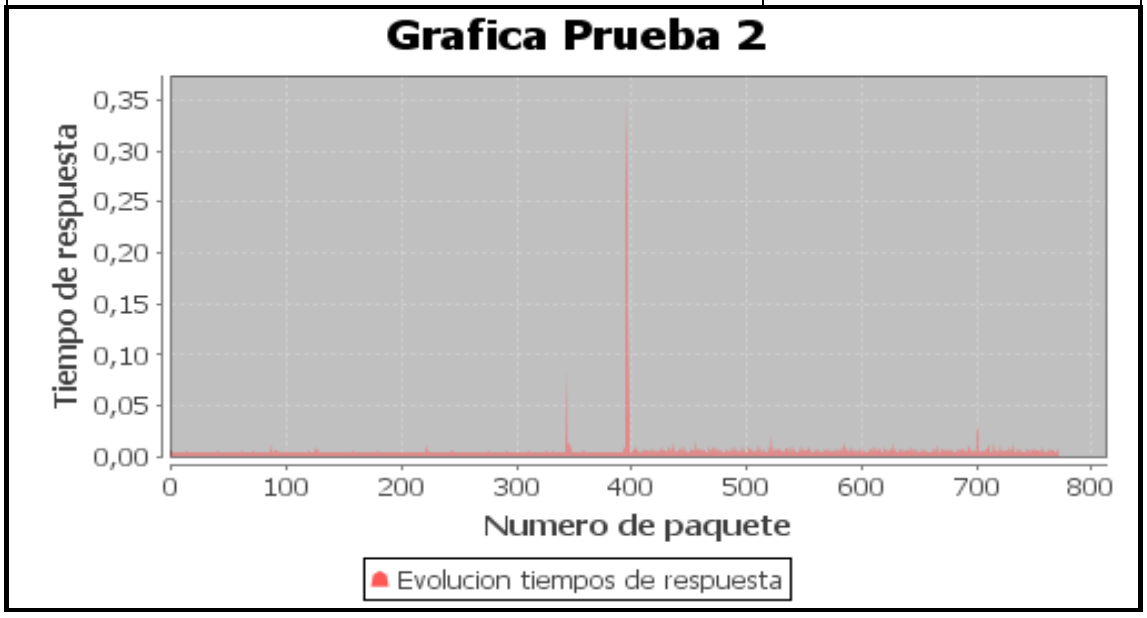
1.2.2.4 Envío de paquetes ping cada 0.1 segundos durante el cambio de red

La última prueba a realizar con la herramienta ping sobre HIP en estructura real es aumentar lo máximo posible la frecuencia de envío de los paquetes, llegando a recibir 10 paquetes por segundo (un paquete cada 0.1 segundos). Dado que ahora están mucho más juntos, se han alargado las pruebas en el tiempo para que se correspondiera con las anteriores; así, en vez de enviar 100 paquetes, se han enviado 1000, y el cambio, en vez de en el paquete 40, se ha intentado realizar en el paquete 400.

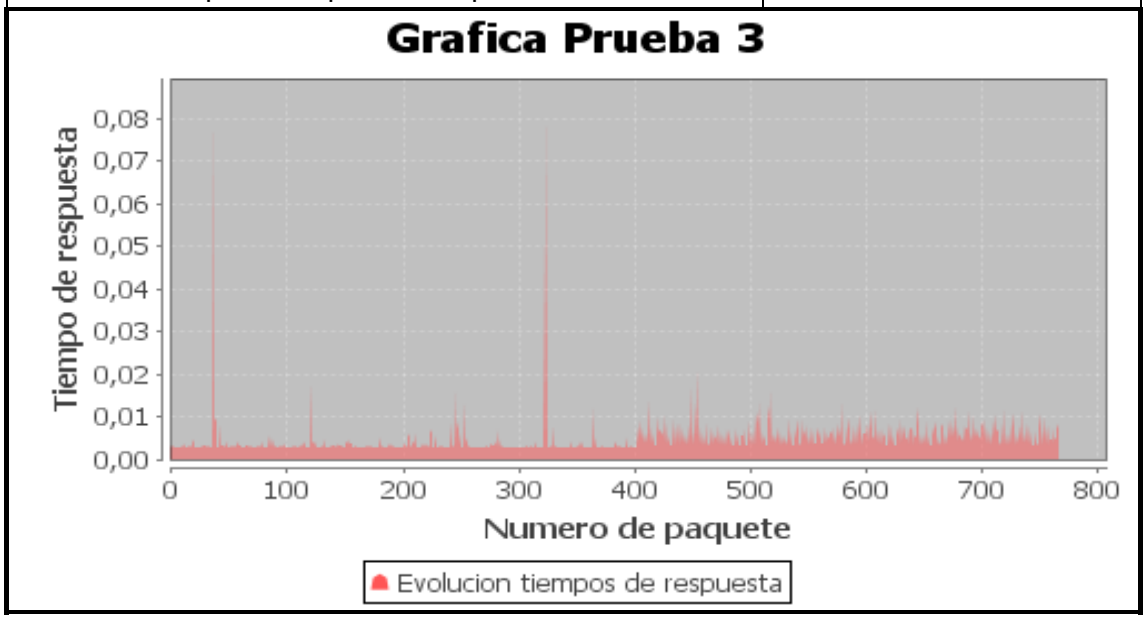
PRUEBA 1.2.2.4.A	
Número total de Request:	**998**
Número total de Reply:	**776**
Número de peticiones perdidas:	**222**
El cambio de red lo efectuó en el paquete:	**403**
Media de tiempo de respuesta antes del cambio:	**0.0037853457 s**
Media de tiempo de respuesta después del cambio:	**0.005902615 s**



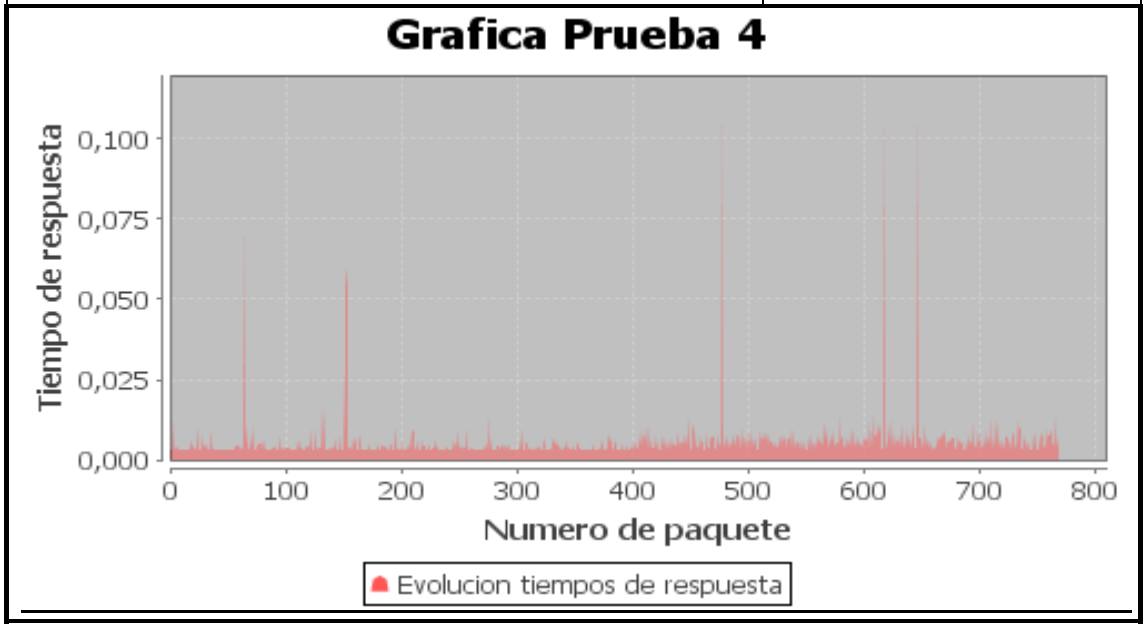
PRUEBA 1.2.2.4.B	
Número total de Request:	**999**
Número total de Reply:	**774**
Número de peticiones perdidas:	**225**
El cambio de red lo efectuó en el paquete:	**402**
Media de tiempo de respuesta antes del cambio:	**0.0052566337 s**
Media de tiempo de respuesta después del cambio:	**0.005293446 s**



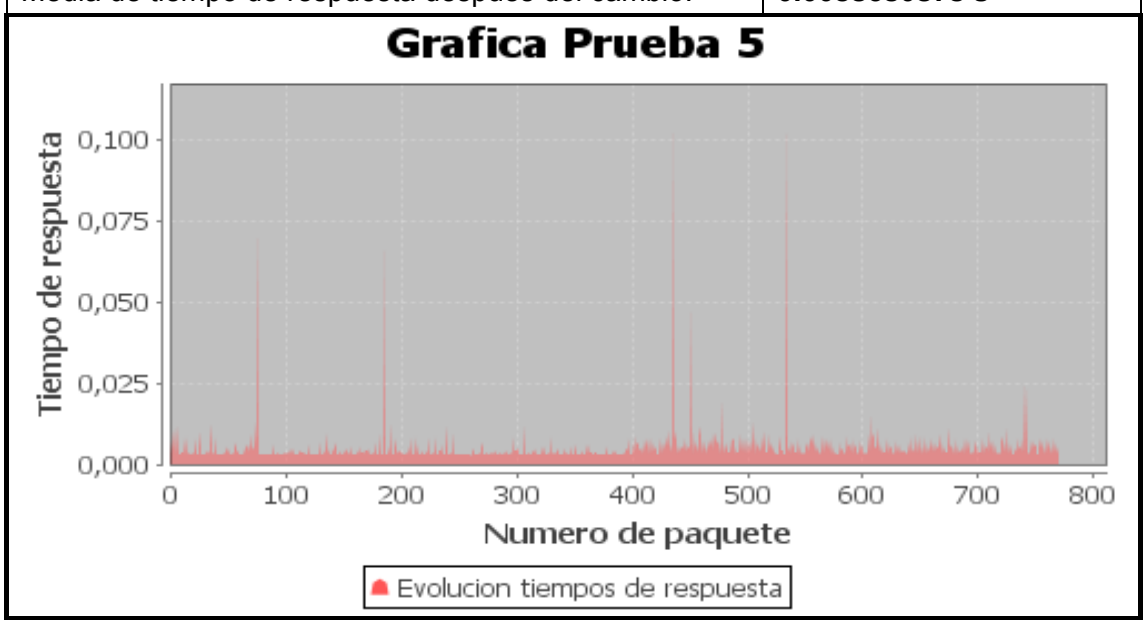
PRUEBA 1.2.2.4.C	
Número total de Request:	**999**
Número total de Reply:	**769**
Número de peticiones perdidas:	**230**
El cambio de red lo efectuó en el paquete:	**403**
Media de tiempo de respuesta antes del cambio:	**0.0037185033 s**
Media de tiempo de respuesta después del cambio:	**0.005650421 s**



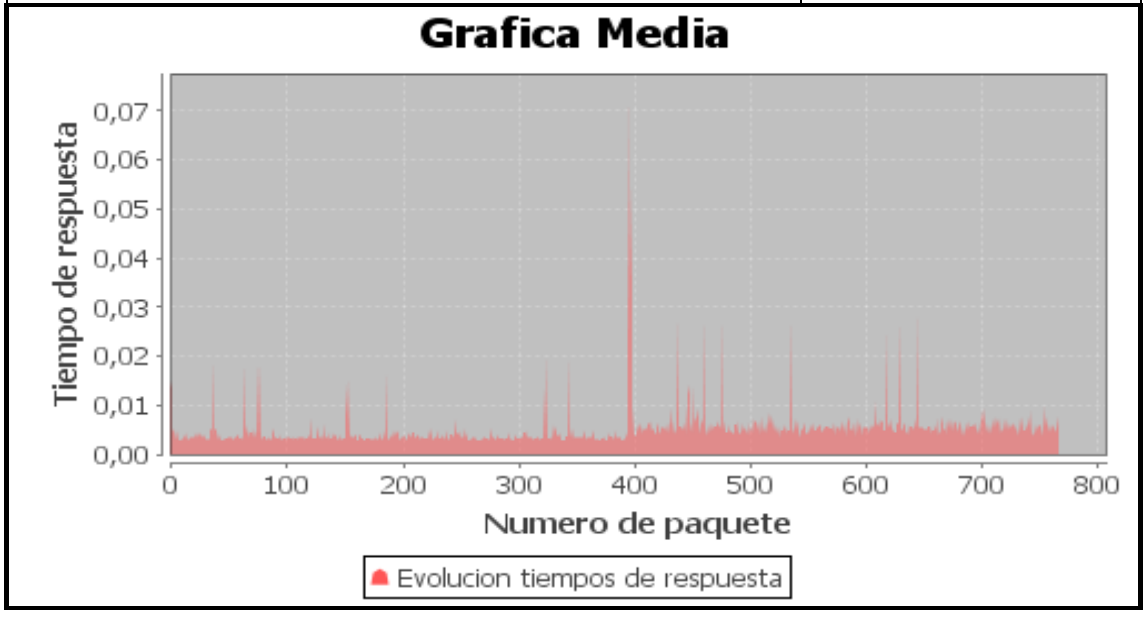
PRUEBA 1.2.2.4.D	
Número total de Request:	**998**
Número total de Reply:	**771**
Número de peticiones perdidas:	**227**
El cambio de red lo efectuó en el paquete:	**400**
Media de tiempo de respuesta antes del cambio:	**0.0042291963 s**
Media de tiempo de respuesta después del cambio:	**0.006453008 s**



PRUEBA 1.2.2.4.E	
Número total de Request:	**997**
Número total de Reply:	**773**
Número de peticiones perdidas:	**224**
El cambio de red lo efectuó en el paquete:	**401**
Media de tiempo de respuesta antes del cambio:	**0.004008647 s**
Media de tiempo de respuesta después del cambio:	**0.0058630873 s**



MEDIA DE LAS PRUEBA 1.2.1.4: PING NORMAL HIP ESTRUCTURA REAL	
El Número de peticiones perdidas:	**225,6**
Media de tiempo de respuesta antes del cambio:	**0,00419960586 s**
Media de tiempo de respuesta después del cambio:	**0,00583251546 s**



Conclusiones

En esta última prueba, y como ya ocurriera con las pruebas que tenían una frecuencia de 0.1, se ha podido ajustar más el tiempo que tarda la conexión en restablecerse. Ahora, el número de paquetes que se ha pedido es de 225.6, y puesto que tenemos una frecuencia de 0.1 segundos entre cada paquete, se observa que el tiempo medio que tarda la conexión en estar activa desde 22.5 segundos, cuando en las pruebas anteriores, como no había una frecuencia que se ajustara tanto, se obtenían resultados cercanos al 23.

En esta prueba, aunque siguen existiendo picos, no es tan problemático para la media como lo podía ser antes, pues el número de paquetes es mucho mayor, y que haya uno un poco desviado no afectará casi a la media. Los resultados obtenidos tanto en red local como en la red a través de HIP son muy similares a los anteriormente obtenidos para otras pruebas, por lo que se puede asegurar que el incremento del número de paquetes por unidad de tiempo a través de HIP con la herramienta ping no afecta al tiempo de recepción de los paquetes.

2.- Peticiones HTTP

Para realizar la prueba de peticiones http, se usó la misma estructura que para las pruebas de ping. En una misma red empiezan el cliente y el servidor, y el cliente comienza a realizar peticiones web sobre el servidor.

Las siguientes figuras ilustran con más detalle la infraestructura construida para esta prueba. En la primera ilustración se puede observar el estado inicial de las máquinas, ambas en la misma red. Cuando se ha llegado a un número de peticiones, se efectúa un cambio de red en el servidor (ilustración 12), y se espera a que se restablezca la conexión, para seguir con las peticiones web, pero con el servidor en una red distinta. Con esto se podrá obtener el tiempo medio de las peticiones a través del protocolo, y el tiempo que tarda en restablecerse la conexión entre el cliente y el servidor.

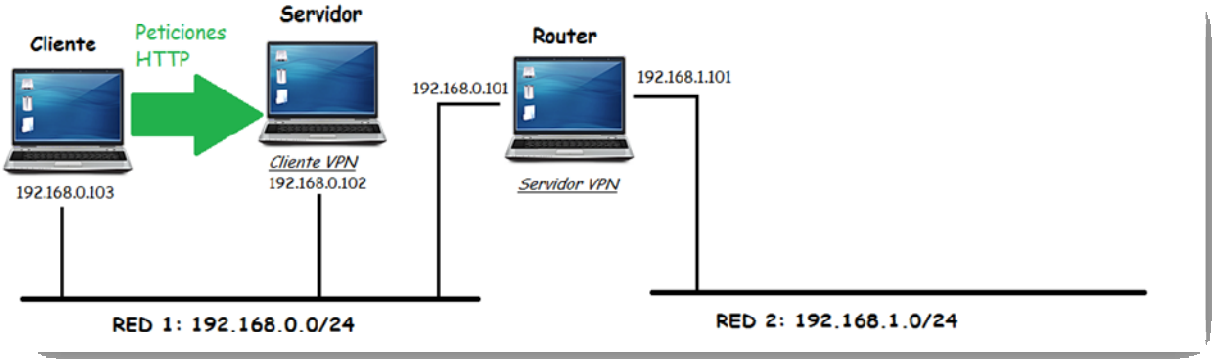


Ilustración 11 : Peticiones http en red local

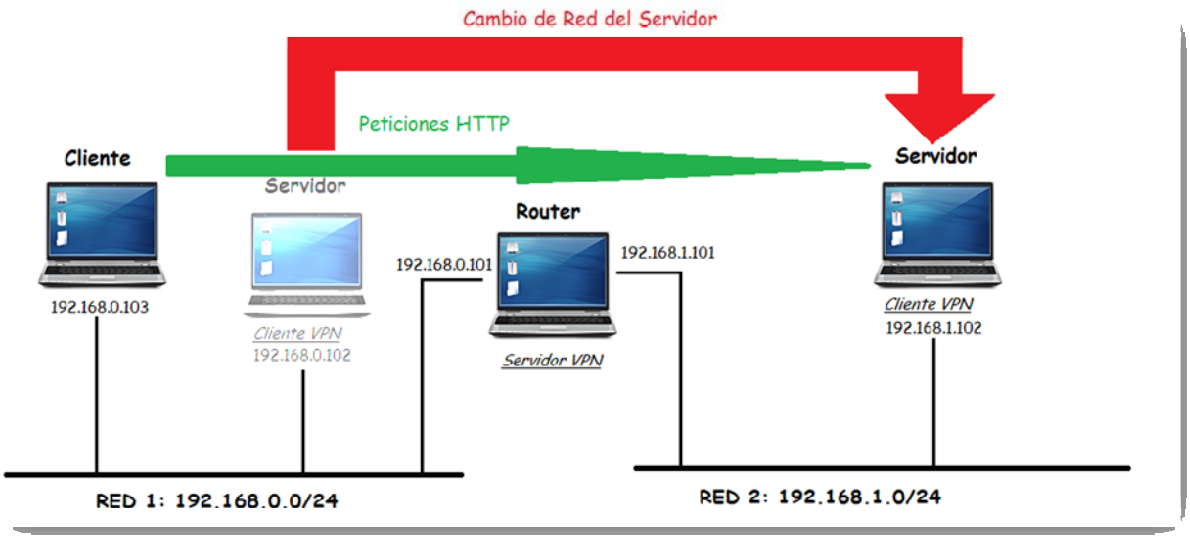


Ilustración 12 : Peticiones http en red pública

2.1- VPN

Descripción de las pruebas

Para realizar las primeras pruebas con Open VPN y un servidor web, se han habilitado tres máquinas. La primera de ellas ha albergado un servidor Open VPN (servidor), la segunda ha tenido el cliente Open VPN y el servidor web (cliente), y el último se ha utilizado para realizar las peticiones HTTP (máquina local).

El objetivo de la prueba era medir el número de peticiones HTTP que se perdían en el momento de cambiar el servidor de una red a otra. Para ello, se ha enviado desde el cliente una petición HTTP al servidor web cada segundo. Sin dejar de hacer dichas peticiones, se ha ejecutado el cambio de red del servidor y se ha esperado un tiempo hasta que se han vuelto a recibir las respuestas de éste en el cliente web. Estas pruebas se han desarrollado en dos entornos diferenciados. Uno ha sido el entorno virtualizado y el otro ha sido en el entorno real, con los ordenadores reales.

Resultados

A continuación se muestran los resultados extraídos de las pruebas descritas anteriormente. En total se han hecho cuatro intentos. Los valores referentes a peticiones enviadas y respondidas han sido iguales en todos los casos, mientras que los tiempos que se muestran en los resultados son una media calculada a partir de todos los intentos:

2.1.1 - Entorno virtual:

Peticiones enviadas: 33
Peticiones perdidas: 2
Peticiones respondidas: 31
Media de tiempo de respuesta antes del cambio: 0.01816174s
Media de tiempo de respuesta después del cambio: 0.031418264s

2.1.2 - Entorno real:

Peticiones enviadas: 33
Peticiones perdidas: 7
Peticiones respondidas: 26
Media de tiempo de respuesta antes del cambio: 0.016256348s
Media de tiempo de respuesta después del cambio: 0.044682682s

Las siguientes gráficas muestran de forma visual los tiempos de respuesta de las peticiones HTTP antes y después de que el servidor web cambie de red. Las primeras cuatro gráficas corresponden a los cuatro intentos, y la última gráfica es el resultado de calcular la media de los tiempos a partir de los cuatro intentos. Todas estas gráficas son referentes a las pruebas en virtual.

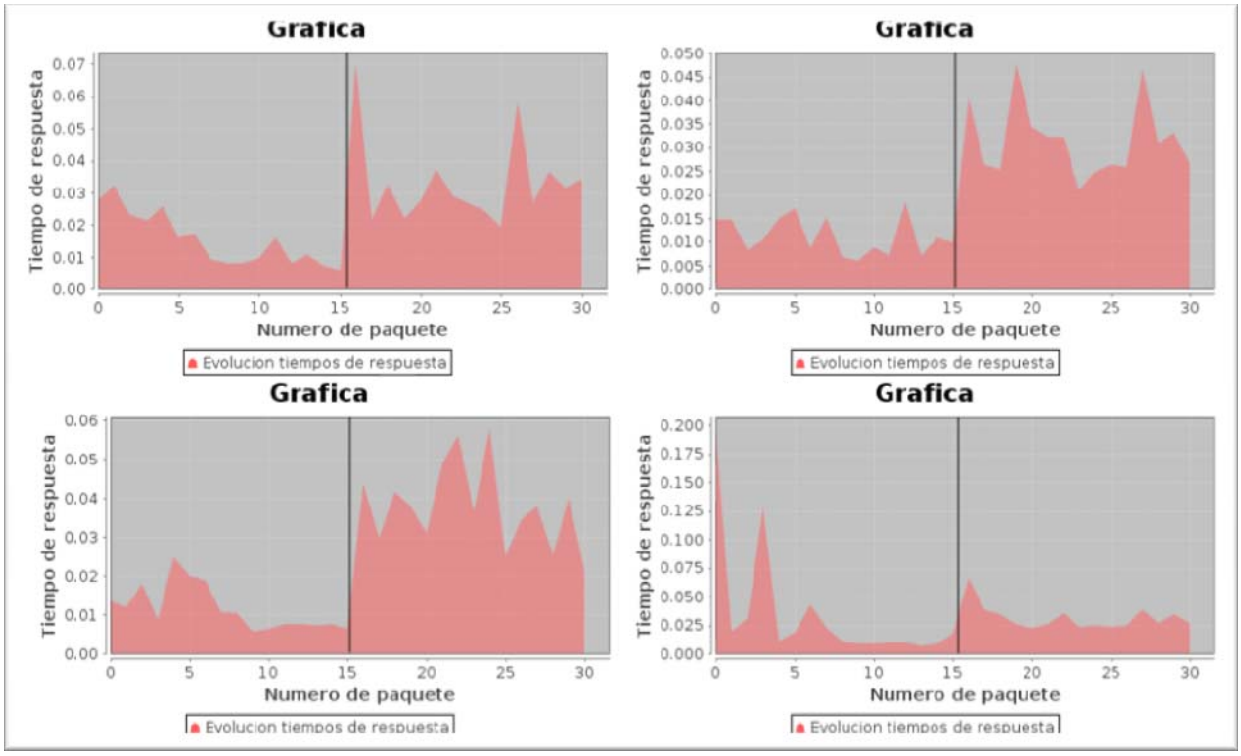


Ilustración 14 : Pruebas peticiones http en local en maquinas virtuales

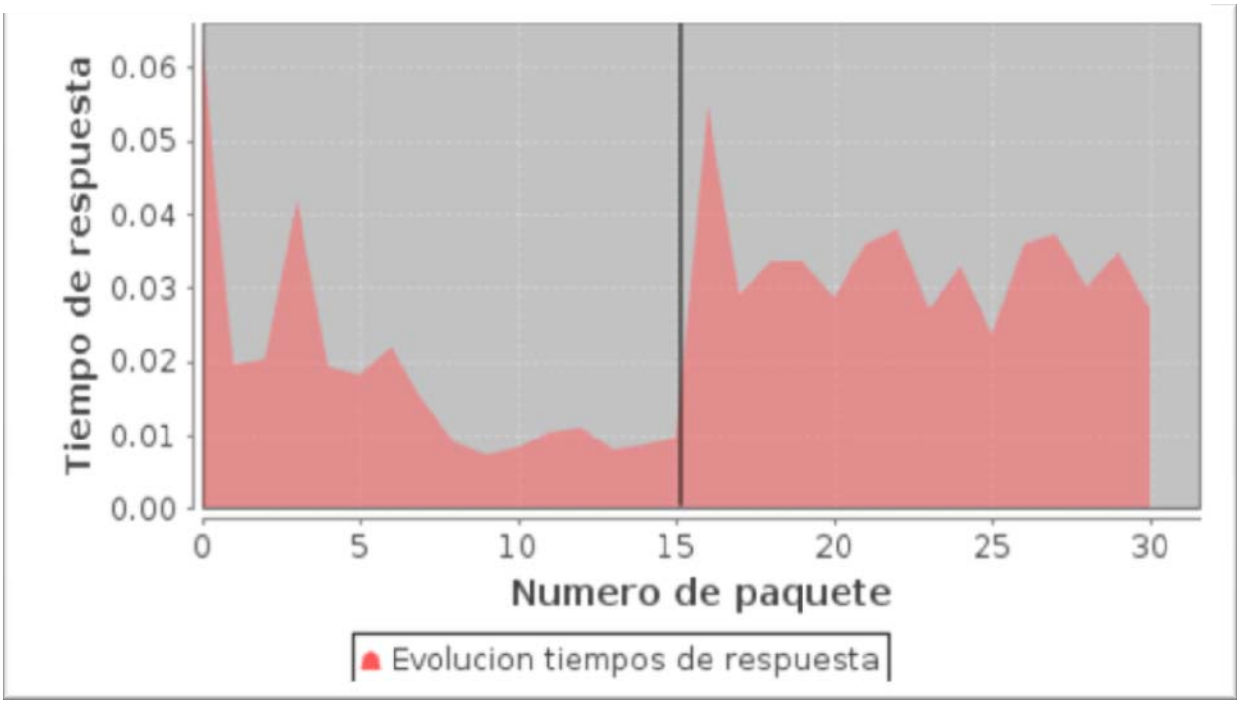


Ilustración 13 : Media pruebas peticiones http en local en máquinas virtuales

De la misma forma que las anteriores, las gráficas que aparecen a continuación corresponden a las mismas pruebas pero esta vez realizadas en el entorno real.

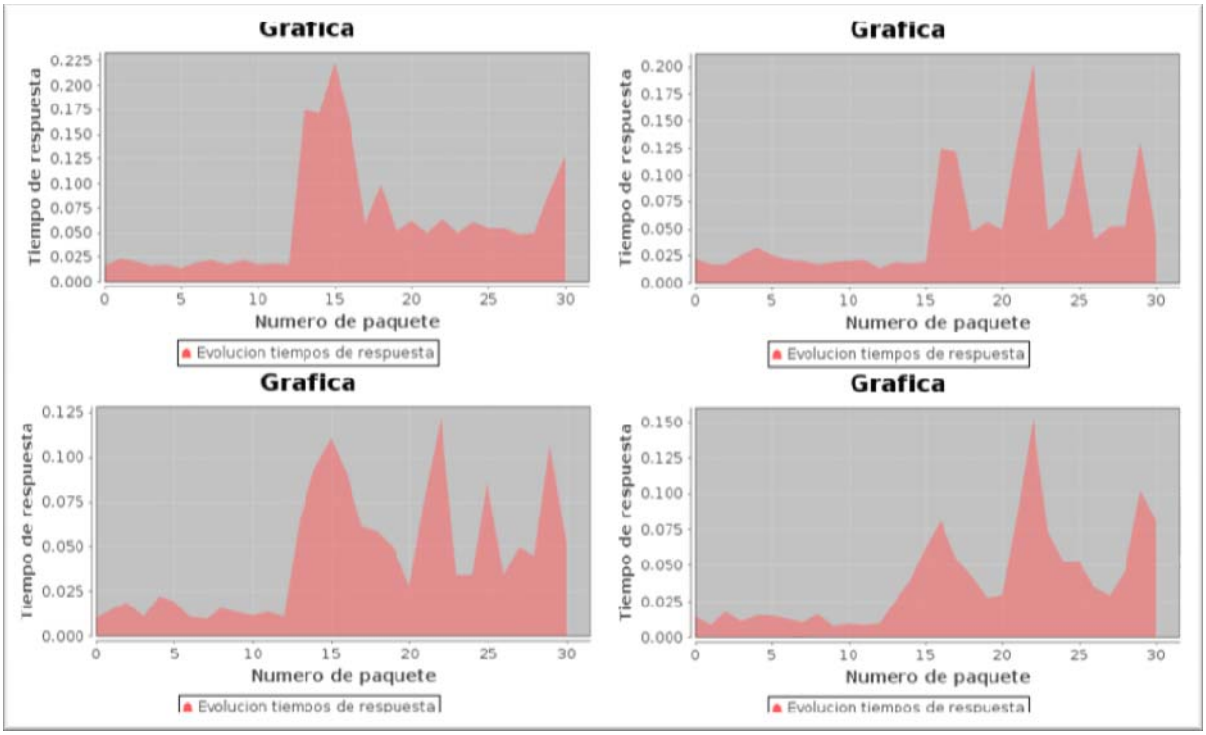


Ilustración 16 : Pruebas peticiones http en real en vpn

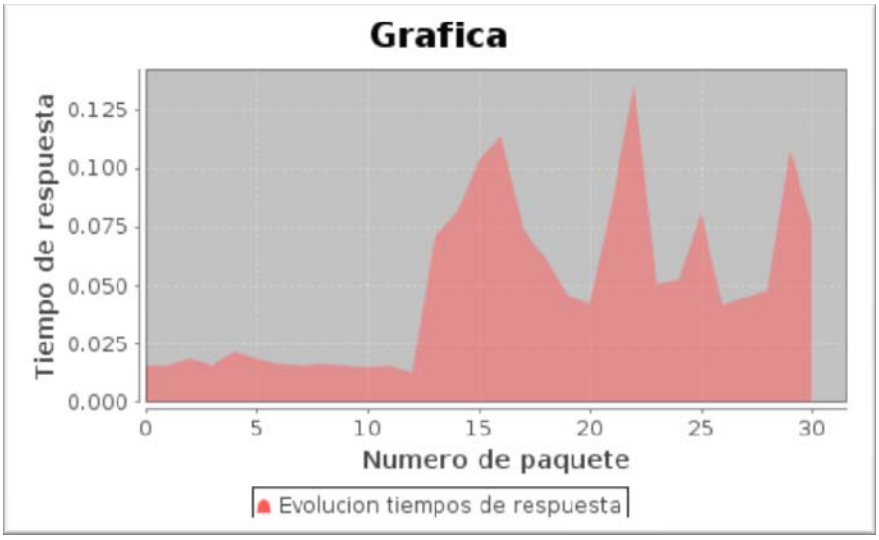


Ilustración 15 : Media pruebas peticiones http en real en vpn

Conclusiones

Una vez realizado este experimento, se han observado algunos hechos que se repiten en todos los casos. Se observa que la primera petición HTTP que se hace una vez realizado el cambio de red por parte del servidor web es considerablemente mayor que todos los siguientes en la mayoría de los casos. Esto es debido a que una vez hecho el cambio de red, y por tanto una vez cambiada la dirección IP del servidor web, el cliente debe hacer una petición ARP para actualizar la información sobre el servidor para poder proseguir con las peticiones.

Por otra parte, se ve de forma clara que el tiempo de respuesta del servidor web es mayor cuando la comunicación se está realizando mediante la tecnología OpenVPN. Esto es un hecho que *a priori* era esperable, ya que Open VPN introduce cierto tráfico adicional para poder hacer su función.

En cuanto a las peticiones que se pierden durante el cambio, el número depende del tiempo que haya entre peticiones. Para el entorno virtual, se ha observado que si el tiempo entre peticiones es de 2 segundos, 1 segundo o medio segundo, durante el cambio se pierden dos peticiones, es decir, que el servidor web deja de recibir dos peticiones del cliente. Sin embargo, si el tiempo entre peticiones es de 3, 4, 5, 6 o 7 segundos, solamente se pierde un paquete durante el cambio. Si el tiempo establecido entre peticiones es de 8 segundos o más, no se pierde ningún paquete, por lo que puede asegurarse que el tiempo que tarda el servidor web en hacer el cambio de red y estar completamente operativo después del cambio se encuentra entre los 7 y los 8 segundos.

En el caso del entorno real, el tiempo de cambio es mucho más grande, ya que dejar una red para ingresar en otra tarda más tiempo. El tiempo de inicio de VPN no varía, pero, como se ha dicho, el tiempo de cambio de red sí que es mayor. El tiempo total de cambio ha rondado entre los 30 y 50 segundos.

2.2- HIP

Descripción de las pruebas

Para la realización de las pruebas se ha montado la infraestructura de dos routers mencionada en la sección infraestructuras creadas para las pruebas.

El servidor se encuentra en la red 192.168.56.0 y es la que se mueve a la red 192.168.57.0.

El cambio de red se ha efectuado ejecutando un script que contiene los siguientes comandos:

```
ifconfig eth0 192.168.57.102
route add default gw 192.168.57.101
```

La máquina servidor tendrá el servidor web instalado y es a ésta a donde hará peticiones web el cliente. La otra máquina es la que mantendrá la conexión de las dos redes en marcha. El cliente y el servidor son las máquinas que contendrán la tecnología HIP.

El objetivo de la prueba era medir el número de peticiones HTTP que se perdían en el momento de cambiar el servidor de una red a otra. Para ello, se ha enviado desde la máquina cliente una petición HTTP al servidor web cada segundo. Sin dejar de hacer dichas peticiones, se ha ejecutado el cambio de red del servidor y se ha esperado un tiempo hasta que se han vuelto a recibir en el servidor las peticiones del cliente.

Resultados

Los resultados del número de peticiones enviadas y recibidas es la misma para las 10 pruebas que se han hecho en entorno virtual. Los tiempos son la media de los 10 intentos; lo mismo para las 10 pruebas realizadas en el entorno real:

2.2.1 - Virtual

Número de peticiones: 37
Número de respuestas: 34
Paquetes perdidos: 3
Media de tiempo de respuesta antes del cambio: 0.06858381
Media de tiempo de respuesta después del cambio: 0.07248476

2.2.2 - Real

Número de peticiones: 33
Número de respuestas: 32
Paquetes perdidos: 1
Media de tiempo de respuesta antes del cambio: 0.022234976
Media de tiempo de respuesta después del cambio: 0.023644209

Las siguientes gráficas muestran de forma visual los tiempos de respuesta de las peticiones HTTP antes y después de que el servidor web cambie de red. Las primeras gráficas corresponden al entorno virtual y las otras 10 al entorno real.

Entorno virtual

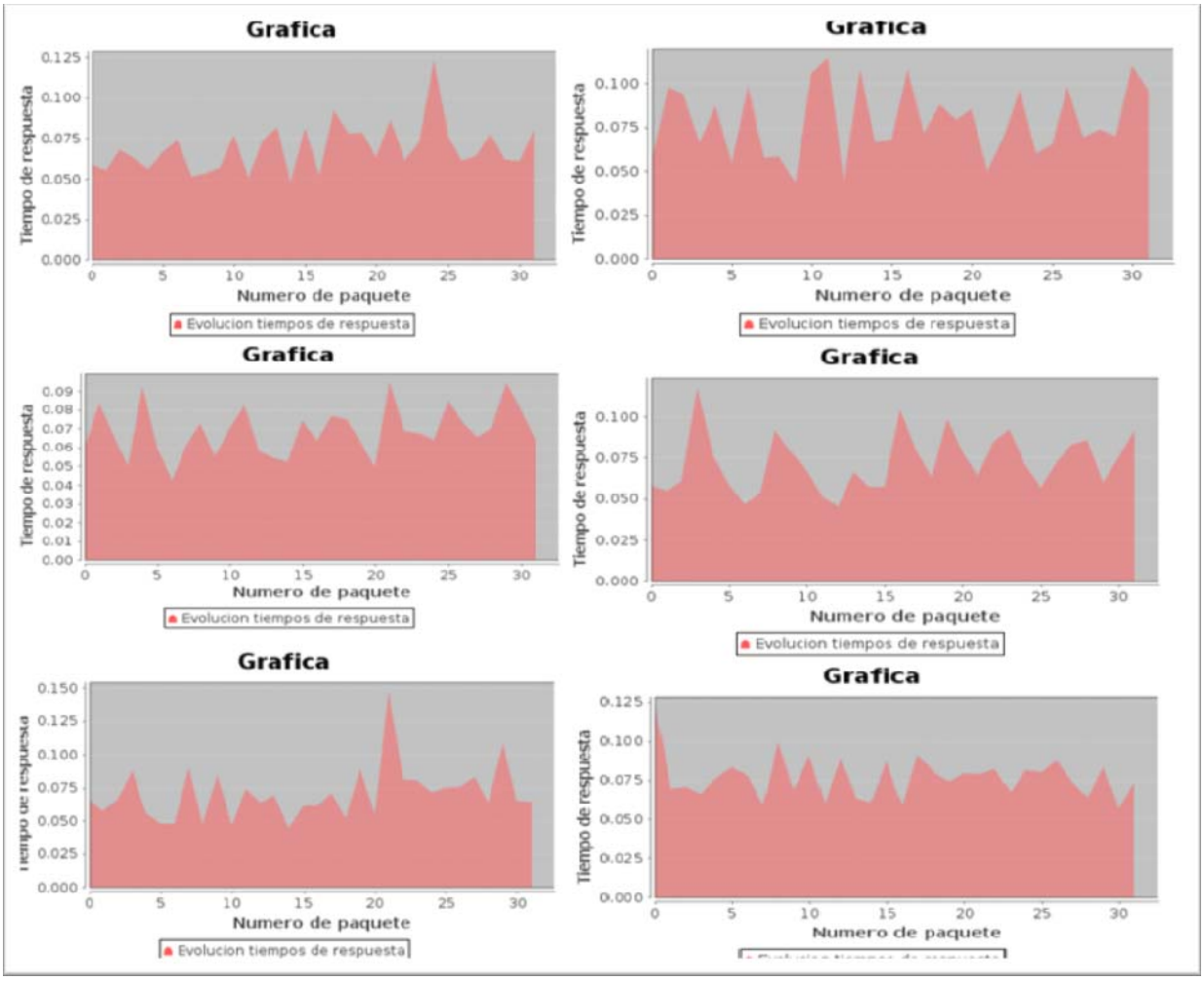


Ilustración 17 : Pruebas peticiones http en maquinas virtuales en hip

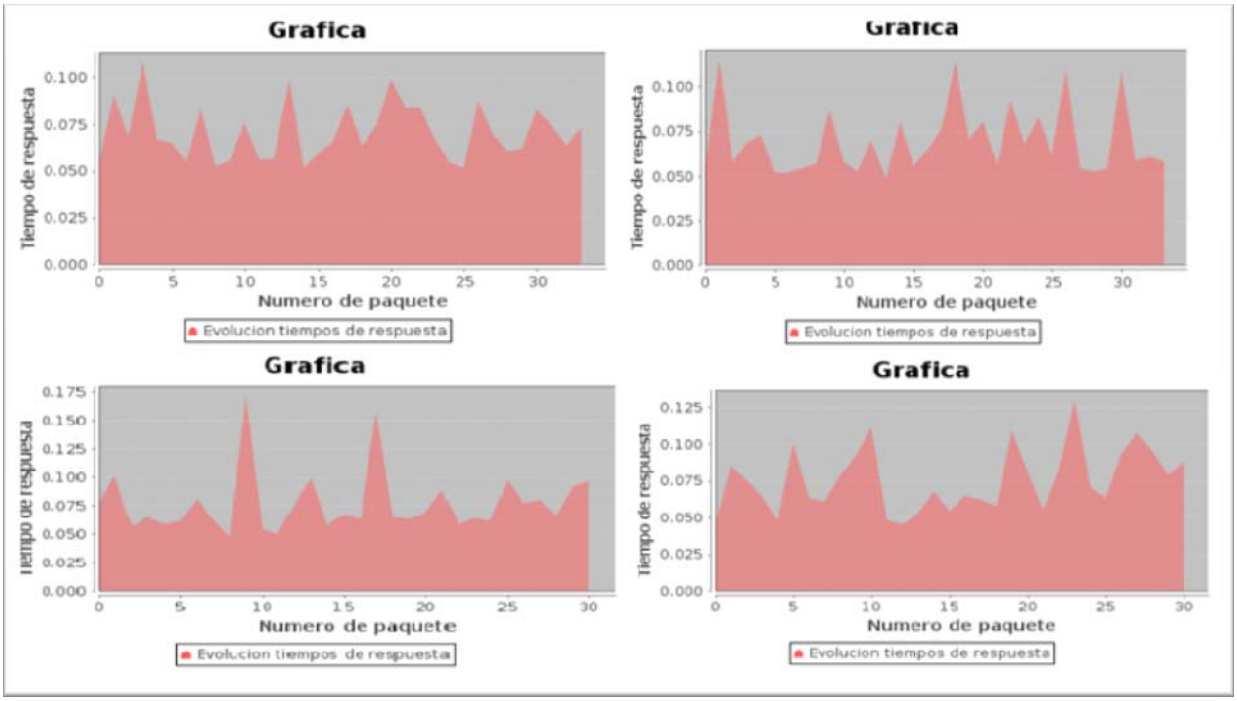


Ilustración 18 : Pruebas peticiones http en maquinas virtuales en hip

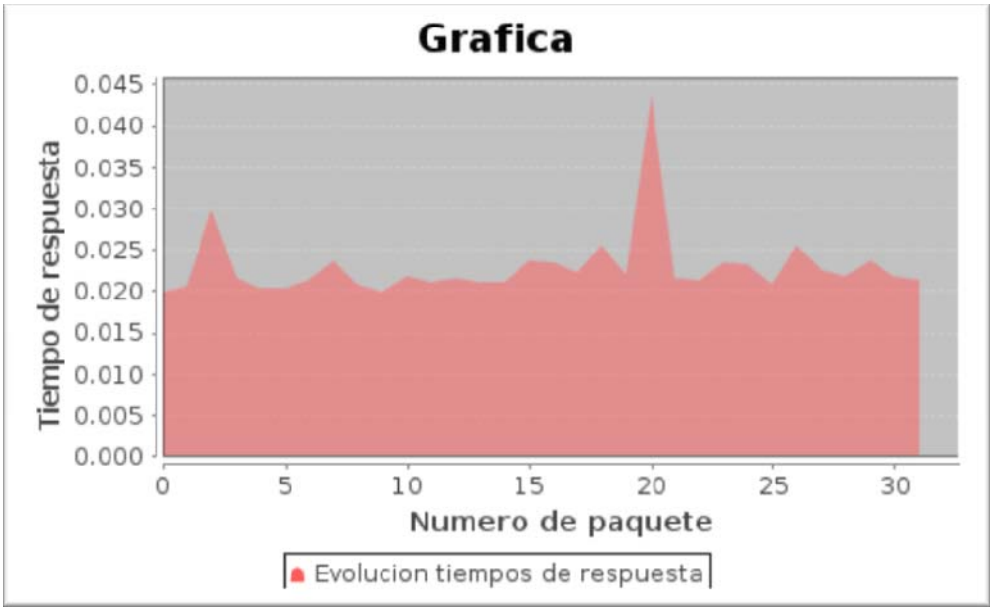


Ilustración 19 : Media pruebas peticiones http en virtual en hip

Y estas 10 pertenecen al **Entorno real:**

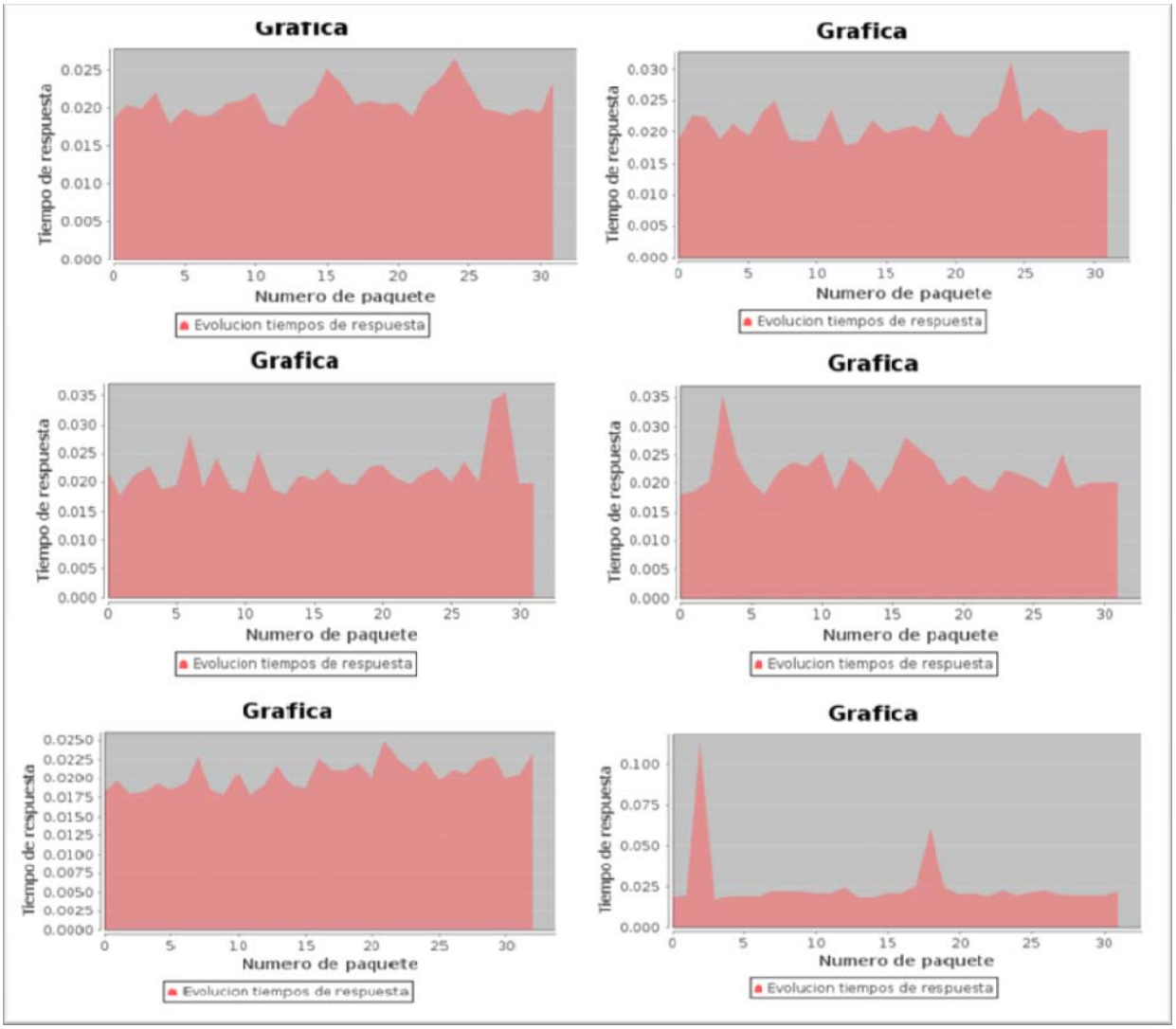


Ilustración 20 : Pruebas peticiones http en real en hip

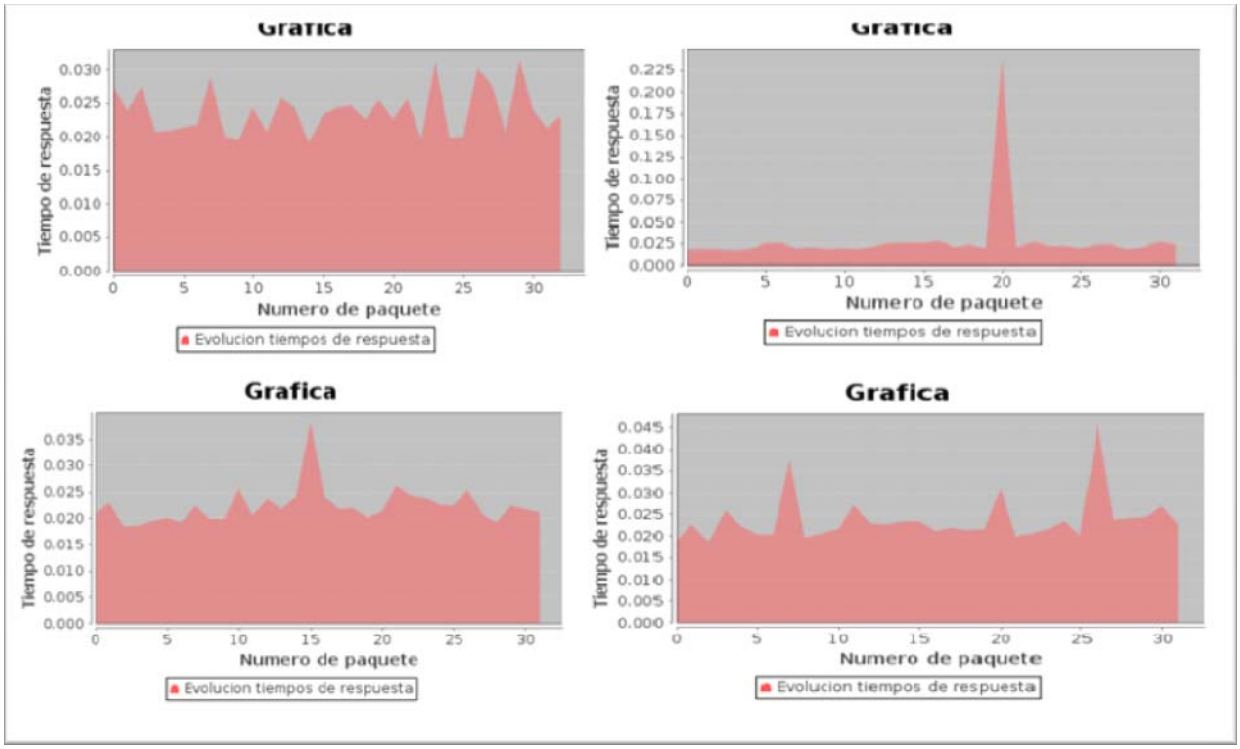


Ilustración 21 : Pruebas peticiones http en real en hip

Ilustración 22 : Media pruebas peticiones http en real en hip

Conclusiones

Si se comparan los paquetes perdidos entre el entorno virtual y el entorno real se puede ver que se han perdido 3 para el virtual y 1 para el real. Un número muy bajo para ambos casos, sabiendo que la reconexión suele tardar unos 20 segundos de media. En lo que a la velocidad de respuesta se refiere, en ambos casos, la respuesta tarda más cuando usa el protocolo HIP, aunque la diferencia es muy pequeña.

Basándose en las gráficas, se puede ver que el tiempo de respuesta de las peticiones es muy parecido todo el rato, la desviación está en torno al 0.05 en el entorno virtual y en el entorno real es considerablemente más pequeño, en torno al 0.02. Existen algunos picos apreciables en el entorno real, pero no le habría que dar demasiada importancia puesto que la conexión era mediante wifi.

3.- Descarga mediante HTTP

Para estudiar la descarga de un fichero mediante http se consideran tres situaciones diferentes. Inicialmente empiezan el cliente y el servidor en la misma red, con el fin de ver la velocidad de la transferencia y el tiempo de descarga cuando el cliente y el servidor se encuentran en la misma red (primera figura).

Después, el servidor se conecta a una nueva red, y se activa uno de los dos protocolos (hip u openvpn), y se obtienen los datos de la transferencia cuando el servidor se encuentra en una red distinta a la del cliente (segunda figura).

Por último, y con el fin de obtener el tiempo que tarda en restablecerse la conexión cuando el cambio de red se efectúa mientras la descarga está activa, se realiza el cambio de red del servidor en un momento dado de la transferencia, con el fin de medir ese tiempo de reconexión.

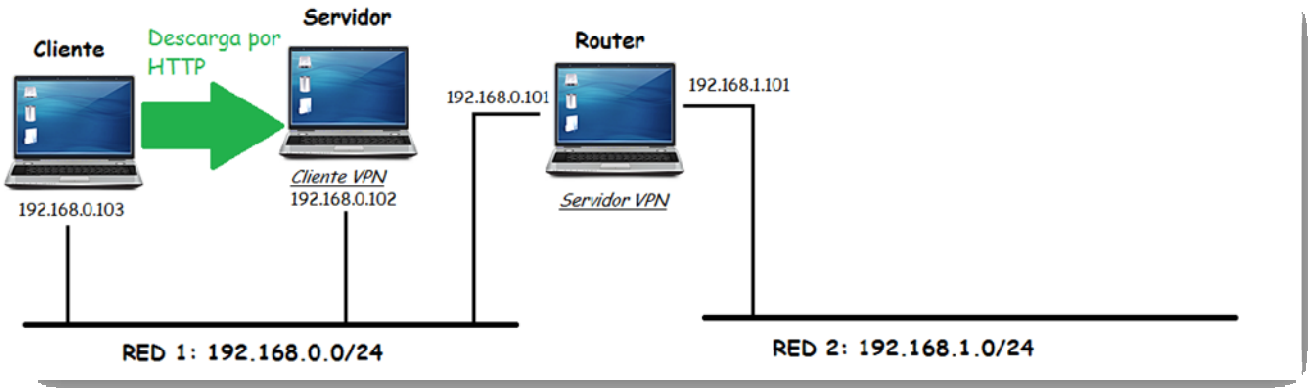


Ilustración 23 : Descarga mediante http en red local

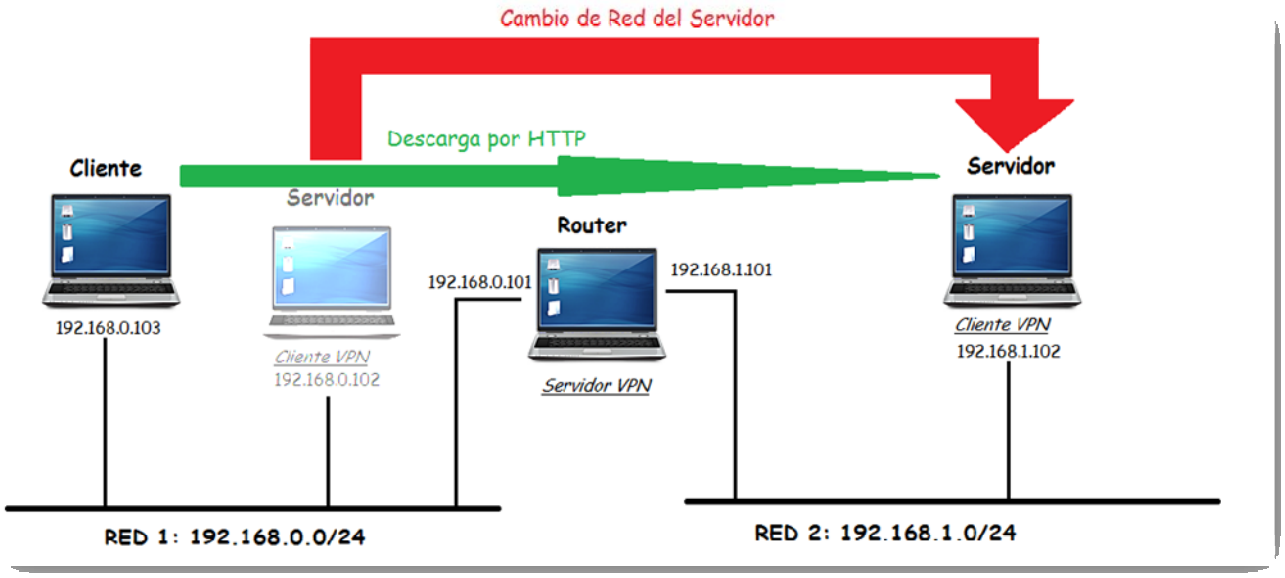


Ilustración 24 : Descarga mediante http en red pública

3.1- VPN

Esta segunda prueba se basa en estudiar el comportamiento de la red a la hora de transferir ficheros de un tamaño considerable. Es posible hacer diferentes intentos dentro de esta segunda prueba, en primer lugar midiendo los tiempos tanto cuando el servidor web se encuentra en la red privada como en la pública y en segundo lugar haciendo el cambio una vez la descarga ya se haya iniciado. En el siguiente apartado se especificarán los resultados obtenidos en ambos tipos de intentos.

3.1.1 Pruebas sin cambiar el servidor de red, descripción de las pruebas

En estos primeros intentos se han medido los tiempos de varias descargas de un fichero un tamaño considerable. Se han utilizado dos ficheros diferentes, uno para las pruebas en virtual y otro para las pruebas en máquinas reales. Se ha hecho esto porque los tiempos en las máquinas reales eran más altos, con lo que el tiempo de espera de las pruebas incrementaba. Para las pruebas en el entorno virtual se ha utilizado un fichero de 231,48 MB, y

para el entorno real ha sido de 22,84 MB. Se han medido las descargas cuando el servidor web se encontraba en la red local y también cuando este se había movido a la red pública y por tanto el tráfico transcurría mediante VPN. Hay que dejar claro que las descargas fueron realizadas completamente sin que el servidor se moviera de una red a otra. En las siguientes tablas se muestran los datos referentes a cada intento, en primer lugar para el servidor web situado en la red privada y en segundo lugar para cuando este estuviera situado en la red pública. Se muestran dichas tablas para los casos de pruebas en máquinas virtuales y en máquinas reales.

3.1.1.1 - Máquinas virtuales

Red privada

Intento	Tiempo de descarga (s)	Velocidad media(MB/s)
1	30	7,71
2	27	8,57
3	17	13,61
4	16	14,47
5	14	16,53
6	13	17,8
7	18	12,86
8	23	10,06
9	24	9,64

Tabla 2 : Descarga mediante HTTP en virtual y local, con vpn

Red Pública

Intento	Tiempo de descarga	Velocidad media(K/s)
1	17m 32s	220
2	18m 8s	212,76
3	17m 55s	215,33
4	17m 57s	214,93
5	19m 34s	197,17
6	18m 20s	210,44

Tabla 3 : Descarga mediante HTTP en virtual y pública, con vpn

3.1.1.2 - Máquinas reales

Red privada

Intento	Tiempo de descarga (s)	Velocidad media(K/s)
1	30	761
2	33	692
3	26	878
4	33	692
5	39	585
6	34	671

Tabla 4 : Descarga mediante HTTP en real y local, con vpn

Red pública

Intento	Tiempo de descarga	Velocidad media(K/s)
1	13m 29s	32
2	11m 34s	30,4
3	12m 15s	18,2
4	10m 41s	33,8
5	10m 43s	30,7
6	10m 31s	46,2

Tabla 5 : Descarga mediante HTTP en real y pública, con vpn

Además de las tablas anteriores, se incluye también una gráfica que muestra de forma más visual los tiempos de descarga de todos los experimentos. En este caso se ha hecho una gráfica para todos los intentos, es decir, aparecen los intentos con el servidor web en la red privada y pública. La primera gráfica corresponde al entorno virtual y la segunda al real. En la primera gráfica el cambio de red se da a partir del noveno intento y en la segunda a partir del sexto.

Ilustración 25 : Media pruebas descarga mediante http en virtual en vpn

Ilustración 26 : Media pruebas descarga mediante http en real en vpn

Conclusiones

Salta a la vista que a la hora de hacer una descarga de un fichero medianamente grande los tiempos de descarga aumentan de forma muy evidente una vez se cambia el servidor de red. Como puede observarse en las gráficas los tiempos de descarga difieren de forma abismal. Esto es debido a que cuando el servidor web está en la red pública todo el tráfico se canaliza a través de VPN, el cual introduce información extra para poder trabajar. Es obvio que el rendimiento de un servidor baja al estar enviando su información usando un VPN.

También se observa que una descarga de un fichero grande es más vulnerable a este tipo de cambios desde el punto de vista de tiempo de descarga, puesto que en el caso de descargar una página web no se aprecia tanto retraso. Esto es debido a que una página web no tiene el tamaño del fichero que se descarga y esto hace que, a pesar de que exista un retardo en la descarga, no se aprecia mucho retraso para cargar la página. Para ver este último hecho solamente es necesario mirar a las dos gráficas expuestas arriba. En la segunda, la perteneciente al entorno real, se ve que las diferencias entre los tiempos en red privada y pública son menores que en el entorno virtual. Esto se debe a que en el caso virtual el fichero era considerablemente más grande. Esto pone de relieve que VPN sufre más cuanto más grande es el volumen de datos que debe transferir.

3.1.2 Pruebas con cambio de red durante la descarga

Descripción de las pruebas

Por último, se han efectuado unas pruebas en las que el servidor web cambia de la red privada a la pública mientras la descarga sigue activa. Los ficheros utilizados para estas pruebas han sido los mismos que en el apartado anterior. El objetivo de esta prueba era principalmente observar el tiempo que se tardaba en tener el sistema listo una vez se comienza a efectuar el cambio de red. Se ha considerado este tiempo como el tiempo transcurrido entre la orden de cambio de red hasta que la descarga ha vuelto a estar activa. Asimismo se han observado las velocidades de descarga tanto antes de efectuar el cambio como después.

Resultados obtenidos

En las siguientes tablas se muestra el tiempo que ha sido necesario desde que se ordena al servidor web cambiar de red hasta que este ha proseguido con el envío del fichero que se estaba descargando, en el entorno virtual y en el real respectivamente.

3.1.2.1 - Entorno virtual

Intento	Tiempo (s)
1	7,9
2	7
3	8,1
4	7,2
5	7,6
6	7,3
7	7,2

Tabla 6 : Tiempos de reconexión de descarga HTTP en virtual con vpn

3.1.2.2 - Entorno real

Intento	Tiempo (s)
1	15.8
2	28.4
3	22.5
4	16.9
5	23.7
6	14.8
7	16.4
8	16.9

Tabla 7 : Tiempos de reconexión de descarga HTTP en real con vpn

Como puede observarse el tiempo de cambio de red oscila entre los 7 y los 8 segundos para el entorno virtual. Bien es cierto que solo se ha medido un intento por encima de los 8 segundos y que el resto se mantiene por debajo de este tiempo. Se observa también que los tiempos tienden más a mantenerse en torno a los 7,2 o 7,3 segundos.

En referencia al entorno con máquinas reales, se observa una mayor diferencia entre los intentos. Hay una diferencia de tiempo de unos 14 segundos entre el más rápido y el más lento. Es cierto que el tiempo de cambio más alto, es excepcional, ya que no se ha observado

ningún otro intento que se le acerque, pero aún eliminando este intento, la variación respecto al entorno virtual es mayor.

Al comparar los valores obtenidos en virtual con los medidos con las máquinas reales, cabe destacar que los tiempos de cambio en la mayoría de los casos son de más del doble para el entorno real. Este es un claro exponente de lo diferente que es un entorno del otro, y de la importancia que tiene el entorno en el que se trabaja a la hora de obtener resultados.

En cuanto a las velocidades de descarga antes y después del cambio de red, tanto en el entorno virtualizado como en el real, no se han encontrado diferencias sustanciales a las mediciones realizadas en los experimentos anteriores, por lo que no se hará ninguna valoración a este respecto.

3.2 - HIP

3.2.1 Pruebas sin cambiar el servidor de red, descripción de las pruebas

En estos primeros intentos se han medido los tiempos de varias descargas de un fichero de un tamaño considerable. Se ha utilizado un fichero de 22.8 MB para estas pruebas. Se han medido las descargas cuando el servidor web se encontraba en la red local (56.0) y también cuando el cliente se había movido de red y por tanto el tráfico transcurría mediante HIP. En las siguientes tablas se muestran los resultados: en primer lugar cuando el cliente se encontraba en la misma red que el servidor y en la segunda cuando el cliente se movía a otra red. Se muestran dichas tablas para los casos de pruebas en máquinas virtuales y en máquinas reales.

Resultados

Los resultados se muestran en forma de tabla con el tiempo que ha tardado el archivo en descargarse y la velocidad media. Se han hecho las pruebas en entorno real y virtual, y para cada uno de estos, cuando el cliente se encontraba en la red 192.168.56.0 (local) y en la red 192.168.57.0 (remota).

3.2.1.1 - Máquinas virtuales

Red local

Intento	Tiempo de descarga (s)	Velocidad media(KB/s)
1	30	889
2	27	985
3	17	1284
4	16	1300
5	14	1342
6	13	1360
7	18	1242
8	23	1051
9	24	1042

Tabla 8 : Descarga mediante HTTP en virtual y local, con hip

Red remota

Intento	Tiempo de descarga(s)	Velocidad Media(KB/s)
1	36	622
2	37	605
3	39	577
4	41	547
5	42	533
6	39	565
7	44	501
8	66	340
9	41	549
10	43	515
Media	42.8	535.4

Tabla 9 : Descarga mediante HTTP en virtual y pública, con hip

3.2.1.2 - Máquinas reales

Red local

Intento	Tiempo de descarga (s)	Velocidad media(K/s)
1	30	761
2	33	692
3	26	878
4	33	692
5	39	585
6	34	671

Tabla 10 : Descarga mediante HTTP en real y local, con hip

Red remota

Intento	Tiempo de descarga(s)	Velocidad media(KB/s)
1	64	347
2	63	356
3	39	569
4	57	391
5	42	531
6	53	420
7	45	494
8	59	376
9	137	163
10	69	322
Media	62.8	396.9

Tabla 11 : Descarga mediante HTTP en real y pública, con hip

Ilustración 27 : Media pruebas descarga mediante http en real en hip

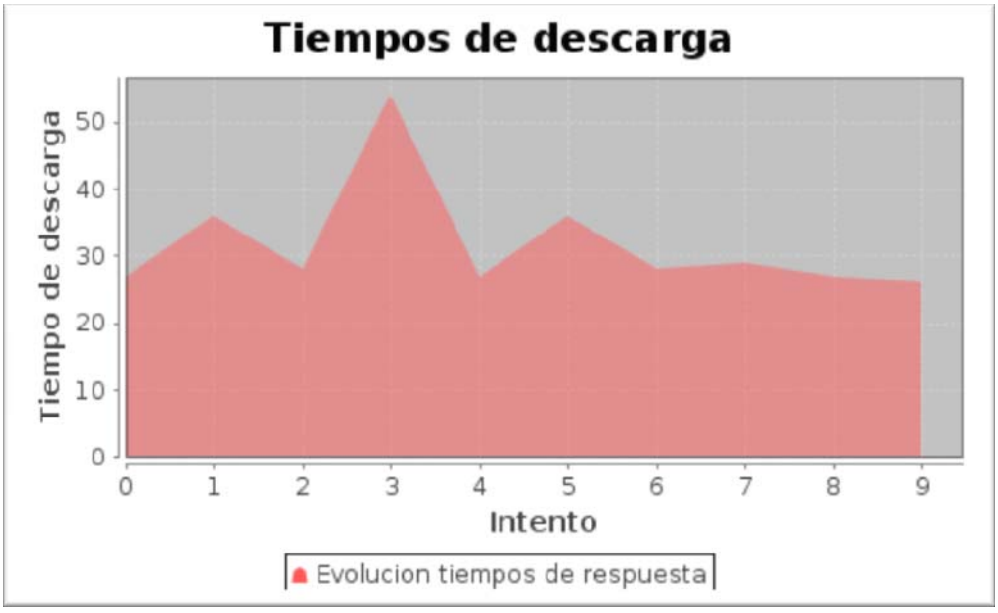


Ilustración 28 : Media pruebas descarga mediante http en virtual en hip

Conclusiones

Lo primero que choca a la vista es lo mucho que baja la velocidad de transmisión cuando el cliente se encuentra en la red 57, bien sea en máquinas virtuales como en la estructura real. Se podría decir que la media de tiempo es el doble bien en estructura real como virtual. Se podría decir también que HIP, la mayoría de las veces, tarda el mismo tiempo y la transferencia es regular. Pero se tiene como excepción dos casos que el tiempo de descarga se dobla. Eso se debe a que los paquetes 'Update' que envía el cliente al cambiar de red no los procesa bien el demonio HIP del servidor.

3.2.2 Pruebas con cambio de red durante la descarga

Por último, se han efectuado unas pruebas en las que el cliente cambia de red mientras la descarga sigue activa. El fichero utilizado para estas pruebas ha sido el mismo que en el apartado anterior. El objetivo de esta prueba era principalmente ver el tiempo que tarda el demonio HIP en darse cuenta que el cliente se ha cambiado de red y medir el tiempo de reconexión. Se ha considerado este tiempo como el tiempo transcurrido entre la orden de cambio de red hasta que la descarga ha vuelto a estar activa. Asimismo se han observado las velocidades de descarga tanto antes de efectuar el cambio como después.

Resultados

3.2.2.1 - Entorno virtual

Intento	Tiempo de cambio(s)
1	21.9
2	13.2
3	24.2
4	12.6
5	16.4
6	12.9
7	16.8
8	18.1
9	6.7
10	18.8
Media	16.16

Tabla 12 : Tiempos de reconexión de descarga HTTP en virtual con hip

3.2.2.2 - Entorno real

Intento	Tiempo de cambio(s)
1	27.1
2	35.6
3	27.5
4	54
5	27.1
6	35.9
7	27.7
8	29.2
9	27
10	25.8
Media	31.69

Tabla 13 : Tiempos de reconexión de descarga HTTP en real con hip

Conclusiones

Como en los casos anteriores, vemos que el funcionamiento del demonio HIP es muy regular menos para algunos casos puntuales. Eso es debido a que HIP no recibe correctamente el paquete UPDATE cuando el cliente se mueve de red y cambia su ubicación y por eso no habilita el cambio de red. Se puede ver que los tiempos son muy parecido en la mayoría de las veces: La media para el entorno virtual está en 16 y sube debido al pico que se ha conseguido de 24 segundos. En el entorno real pasa lo mismo: Se obtienen unos tiempos de entre 26 y 30 pero a veces el demonio HIP falla y sube la reconexión a los 35 incluso 54 segundos.

4.- SCP

A continuación se adjuntan las imágenes que ilustran la situación de la estructura de las máquinas para la realización de esta prueba.

Inicialmente empiezan el cliente y el servidor en la misma red, y comienza la transferencia del fichero del servidor al cliente, obteniendo el tiempo que tarda en efectuarse dicha transferencia. Esto queda reflejado en la primera ilustración.

Después, el servidor cambia de red, y se realiza la misma prueba, para obtener los tiempos de transferencia cuando el servidor se encuentra en otra red y los paquetes viajan a través del protocolo. Esta situación se corresponde con la segunda ilustración.

Por último, se inicia la transferencia desde la red inicial, y en un momento dado de la misma, se migra el servidor a otra red. La transferencia vuelve a reactivarse un tiempo después de restablecerse la conexión del servidor a la nueva red. El objetivo de esta tercera prueba es medir ese tiempo.

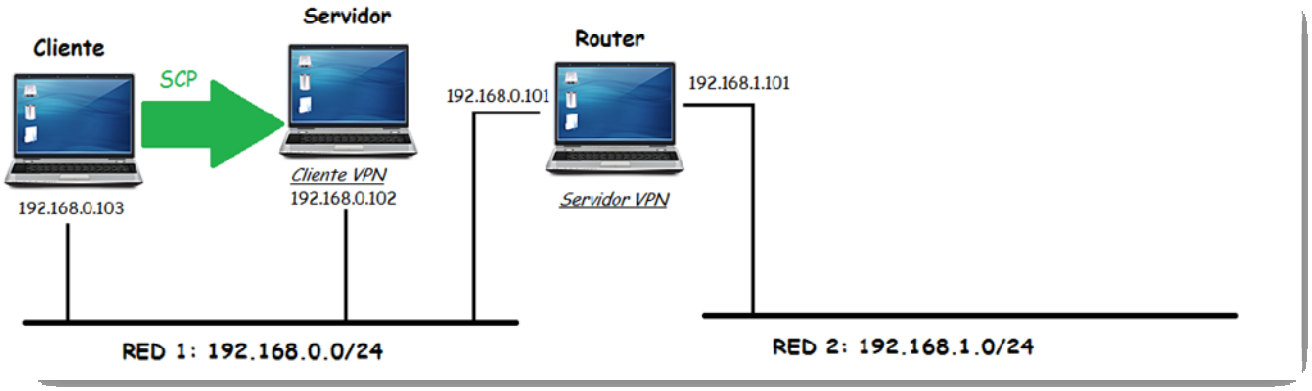


Ilustración 29 : SCP en red local

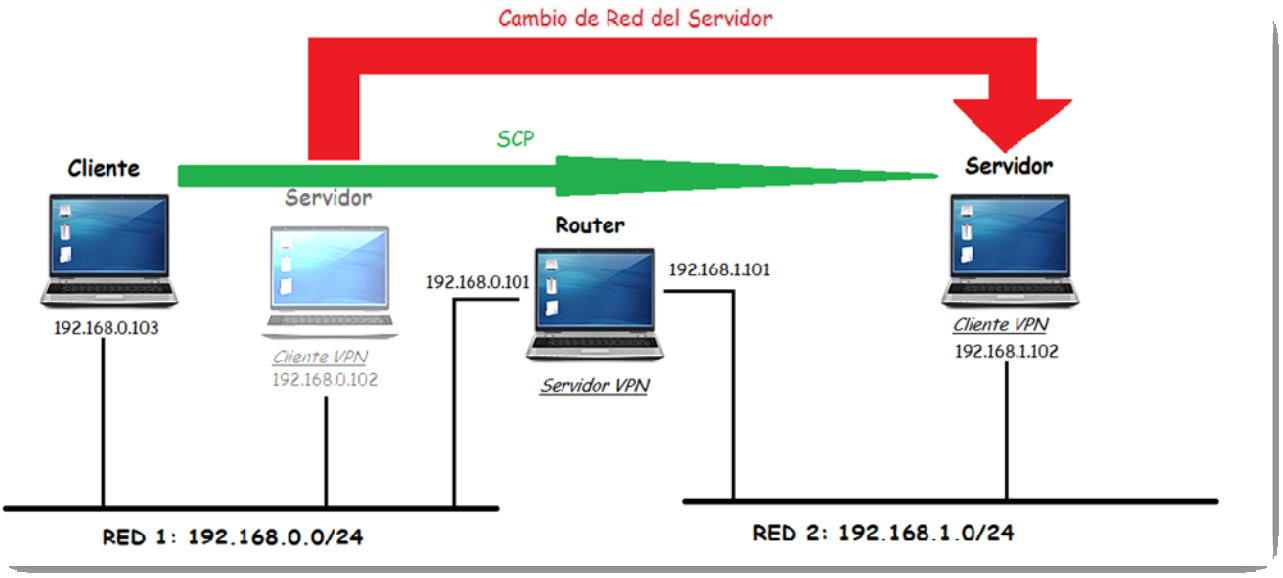


Ilustración 30 : SCP en red pública

4.1 - VPN

Otra prueba que se ha realizado sobre OpenVPN ha sido la transferencia de un fichero relativamente grande, como en uno de los anteriores apartados, pero con la diferencia de que en este caso se ha utilizado el comando scp para efectuar la transferencia, en lugar de hacerlo utilizando un servidor web y el comando wget. En este caso, nuevamente, se han hecho diferentes tipos de pruebas. En primer lugar se han hecho varias transferencias del fichero sin hacer ningún tipo de modificación a la infraestructura utilizada, y estando el ordenador emisor en la misma red que el receptor. El segundo tipo de pruebas han sido iguales que las primeras, con la diferencia de que la máquina emisora del fichero estaba en una red diferente a la del receptor, estando estas visibles la una de la otra gracias a un servidor VPN que daba una IP de la red en la que estaba el ordenador receptor, al ordenador emisor. Finalmente, se han efectuado unas pruebas haciendo el cambio de red durante la transferencia del fichero, por lo que la transferencia comenzaba siendo entre dos ordenadores en una misma red y finalizaba teniendo que atravesar una infraestructura VPN. En los siguientes apartados se explicarán más en detalle las pruebas, y se darán los resultados de las mismas.

4.1.1 - Transferencia en red privada

Descripción

Como se ha explicado en la introducción de esta sección de la memoria, las primeras pruebas con scp han consistido en transferir un fichero de 221 MB entre dos computadores situados en la misma red, una red que se llamará red privada. En este caso, se ha utilizado el mismo fichero tanto para el entorno virtual como para el real, ya que se ha observado que el tiempo no era excesivamente alto en el segundo caso. El objetivo de dichas pruebas era el de evaluar el comportamiento del comando scp en un entorno en el que la transferencia se hiciera directamente de una máquina a otra. Esta evaluación servirá más a delante para comparar los resultados obtenidos cuando la transferencia se hace utilizando un servidor VPN.

La evaluación ha consistido en medir el tiempo necesario para la transferencia del mencionado fichero y la media de velocidad a la que se ha enviado de un ordenador a otro.

Resultados obtenidos

En las siguientes tablas se muestran los valores medidos en las pruebas descritas en el anterior apartado, en primer lugar para el entorno virtualizado y en segundo lugar para las máquinas reales. La tabla muestra el tiempo necesario y la velocidad de transferencia en todos los intentos realizados.

4.1.1.1 - Entorno virtual

Intento	Tiempo de descaga (s)	Velocidad media (MB/s)
1	38	5,8
2	30	7,4
3	24	9,2
4	22	10
5	21	10,5
6	22	10
7	21	10,5
8	20	11
9	24	9,2
10	15	14,7
11	14	15,8

12	16	13,8
13	19	11,6
14	24	9,2
Media	19,28	10,56

Tabla 14 : SCP en virtual y local, con vpn

4.1.1.2 - Entorno real

Intento	Tiempo de descarga	Velocidad media (MB/s)
1	02:03	1.8
2	02:03	1.8
3	02:02	1.8
4	02:01	1.8
5	02:02	1.8
6	02:07	1.7
7	03:01	1.2
8	02:47	1.3
9	02:15	1.6
10	02:00	1.8
11	01:59	1.9
Media	02:12	1.68

Tabla 15 : SCP en real y local, con vpn

Conclusiones

Como se puede ver en la tabla anterior, los tiempos de descarga, así como las velocidades de transferencia han sido bastante homogéneos en todos los intentos, habiendo rondado los 19 segundos para completar la operación en el entorno virtual y aproximadamente dos minutos y algunos segundos en el entorno real. Cabe decir que las mayores diferencias se han dado en este caso en las pruebas en virtual, ya que hay cuatro intentos en los que los

tiempos han sido diferentes al resto, muy altos en dos casos y muy bajos en otros dos. En el caso de los primeros el tiempo de descarga ha subido por encima de los treinta segundos, mientras que en el caso de los segundos no ha superado los 15 segundos. La diferencia de tiempos y velocidad en estos cuatro casos, y la homogeneidad del resto en unos valores bastante similares, invita a restar valor a los resultados obtenidos en ellos.

De esta misma forma, hay dos intentos en el caso de las máquinas reales en los que el tiempo se ha disparado hasta rondar los tres minutos (intentos 7 y 8). El resto de los resultados se han mantenido muy similares, con una diferencia de unos pocos segundos entre unos y otros.

4.1.2 - Transferencia en red pública

Descripción

La segunda prueba realizada con el comando scp, como bien se ha comentado con anterioridad, ha consistido en realizar una transferencia de un fichero de 221 MB desde un ordenador situado en la red privada y otro en la red pública. Concretamente, es el ordenador emisor el que se sitúa en la red pública, necesitando de un servidor VPN que le proporcione una dirección IP de la red privada. En este segundo caso, el principal objetivo es comparar los tiempos obtenidos en las anteriores pruebas para estudiar los posibles retardos introducidos por el protocolo VPN.

Como en los anteriores casos, las pruebas se han realizado en entorno virtual y real.

Resultados obtenidos

En las siguientes tablas se muestran los tiempos de descarga y la velocidad de transferencia obtenidos en la situación descrita anteriormente. La primera tabla corresponde al entorno virtualizado y la segunda al real.

4.1.2.1 - Entorno virtual

Intento	Tiempo de descarga	Velocidad media (KB/s)
1	25:53	145,6
2	23:13	162,3
3	24:03	156,7
4	23:01	163,7
5	22:33	167,1
6	23:24	161,0
7	22:46	165.8
8	21:59	171.6
9	24:18	155.6
10	23:41	159.6
Media	21:30	161.44

Tabla 16 : SCP en virtual y pública, con vpn

4.1.2.2 - Entorno real

Intento	Tiempo de descarga	Velocidad media (KB/s)
1	41:50	90.1
2	42:07	89.5
3	39:36	95.1
4	42:23	89.0
5	40:47	92.4
6	41:35	90.6
7	39:58	94.2
8	41:35	90.6
9	39:40	94.9
10	41:26	90.9
Media	41:05	91.73

Tabla 17 : SCP en real y pública, con vpn

Conclusiones

En este segundo conjunto de pruebas con scp, los cuales se han realizado a través de un servidor VPN, muestran un notorio aumento del tiempo necesario para completar la descarga. Ninguno de los intentos realizados ha bajado de los 22 minutos, mientras que el intento que mayor tiempo ha necesitado en el conjunto de pruebas anteriores no ha llegado a los cuarenta segundos. La principal razón de este descenso en la velocidad de transferencia es sin lugar a dudas la necesidad de utilizar un VPN, que introduce información extra en las transferencias para poder llevar a cabo las redirecciones pertinentes y tomar las medidas de seguridad necesarias para el correcto funcionamiento de la infraestructura. A falta de realizar todas las pruebas, parece claro que la movilidad con VPN tiene un gran escollo que imposibilita su uso en muchos casos de la vida real, y no es otro que la baja tasa de velocidad obtenida. Esto es más patente si cabe cuando las transferencias corresponden a ficheros grandes que hay que separar en muchos paquetes, ya que cada paquete supone un porcentaje de pérdida de eficiencia, y por tanto, cuantos más paquetes se necesiten menor será la eficiencia lograda.

La comparación entre los resultados en virtual con los mismos obtenidos de las máquinas reales muestra claramente que en el caso del entorno virtual la transferencia es mucho más rápida. De hecho, puede decirse que los tiempos se reducen a la mitad en el caso del entorno virtualizado, lo cual enseña que en una situación de uso real de este sistema, se estaría ante un problema grave desde el punto de vista de velocidades de transferencia.

4.1.3 - Cambio de red durante la transferencia

Descripción

Para finalizar con el comando scp, se ha efectuado un tipo de pruebas en las que se ha efectuado un cambio de red del ordenador emisor del fichero, desde la red privada hasta la red pública. El objetivo principal de esta prueba ha sido el de medir el tiempo que se tarda en realizar el cambio de red, es decir, el tiempo que se tarda desde que, estando la transferencia en curso, se inicia el proceso de cambio de red por parte del emisor, hasta que la transferencia vuelve a estar completamente operativa. Este proceso, requiere poner en marcha las interfaces de red de la red pública, desactivar las interfaces de red de la red privada y poner en marcha el VPN. En segundo lugar, es interesante evaluar las velocidades de transferencia teniendo al emisor tanto en la red privada como en la pública. De esta forma se podrá comparar la velocidad de transferencia conseguida en cada uno de estos casos con las velocidades de las pruebas anteriores, y por tanto, ver si el propio hecho de cambiar el ordenador a la red pública afecta a las velocidades de transferencia obtenidas en los anteriores casos.

Resultados obtenidos

En las siguientes tablas se muestran los tiempos necesitados por el sistema para hacer el cambio del ordenador emisor desde la red privada a la pública. La primera tabla especifica los resultados obtenidos con la infraestructura virtual, y la segunda con las máquinas reales.

4.1.3.1 - Entorno virtual

Intento	Tiempo de cambio (S)
1	12,9
2	13,1
3	16,2
4	20
5	12,8
6	10
7	12,3
8	8,7
9	12,9
10	13,1
11	11,1
Media	13,009

Tabla 18 : Tiempos de reconexión de SCP en virtual con vpn

4.1.3.2 - Entorno real

Intento	Tiempo de cambio
1	0:44
2	2:05
3	2:20
4	1:41
5	1:50
6	2:13
7	1:36
8	1:58
9	1:30
10	2:18
Media	1:49

Tabla 19 : Tiempos de reconexión de SCP en real con vpn

Conclusiones

Como puede observarse en la anteriores tablas, el tiempo de cambio del emisor de la red pública a la red privada tiene una media de unos 13 segundos en virtual y casi dos minutos en real. Hay que resaltar que los tiempos no han sido demasiado homogéneos, puesto que la variación de tiempos entre intentos se sitúa en una franja de 12 segundos para el entorno virtual y un minuto 36 segundos en el real (diferencia entre el menor tiempo y el mayor). También es cierto que el mayor y el menor tiempo medidos han sido valores extraordinarios en ambos casos, y que el resto han estado entre los 16 y los 10 segundos para en virtual y 1:40 - 2:00 en real. Aún así, estas diferencias pueden considerarse algo heterogéneas, dadas las diferencias entre intentos.

Las conclusiones que deja este último tipo de pruebas es que es necesario cierto tiempo para realizar el cambio de una red a otra y volver a tener totalmente operativa la transferencia, sobre todo en el entorno real. En este último caso, los tiempos de cambio de red son inasumibles para casi cualquier tipo de tarea, ya que para necesidades en las que el tiempo es un factor crítico este cambio de red podría suponer demasiado tiempo con la red inoperativa.

En las tablas anteriores solamente se han mostrado los valores referentes al cambio de red, a pesar de que en la descripción de la prueba se mencionaba también que sería interesante medir las velocidades de transferencia. Esta información no se ha introducido en este apartado porque las velocidades obtenidas en la red privada eran muy similares a las obtenidas en el primer tipo de pruebas, y las velocidades en la red pública, por su parte, muy parecidas a las del segundo tipo de pruebas.

4.2 - HIP

La tercera prueba que se ha realizado sobre HIP ha sido la transferencia de un fichero de 221 MB, como en el anterior apartado, pero con la diferencia de que en este caso se ha utilizado el comando scp para efectuar la transferencia. Se han hecho diferentes tipos de pruebas. En primer lugar se han hecho varias transferencias del fichero sin que el cliente cambiase de red. El segundo tipo de pruebas han sido iguales que las primeras, con la diferencia de que el cliente estaba en una red diferente a la del servidor, estando éstos visibles el uno del otro debido a la tercera máquina que hacía de encaminador. Finalmente, se han efectuado unas pruebas haciendo el cambio de red durante la transferencia del fichero. En los siguientes apartados se explicarán más en detalle las pruebas, y se darán los resultados de las mismas.

4.2.1 - Transferencia en misma red

Descripción

Como se ha explicado en la introducción de esta sección de la memoria, las primeras pruebas con scp han consistido en transferir un fichero de 221 MB entre dos computadores situados en la misma red. El objetivo de dichas pruebas era el de evaluar el comportamiento del comando scp en un entorno en el que la transferencia se hiciera directamente de una máquina a otra. Esta evaluación servirá más a delante para comparar los resultados obtenidos cuando la transferencia se hace utilizando HIP.

La evaluación ha consistido en medir el tiempo necesario para la transferencia del mencionado fichero y la media de velocidad a la que se ha enviado de un ordenador a otro.

Resultados obtenidos

En las siguientes tablas se muestran los valores medidos en las pruebas descritas en el anterior apartado, en primer lugar para el entorno virtualizado y en segundo lugar para las máquinas reales. La tabla muestra el tiempo necesario y la velocidad de transferencia en todos los intentos realizados.

4.2.1.1 - Entorno Virtual

Intento	Tiempo de descaga (m)	Velocidad media (KB/s)
1	5	1000
2	5.33	948.8
3	6.52	766.9
4	5.52	897.6
5	8.27	623.2
6	5.38	934.7
7	7.15	726.3
8	7.23	713.2
9	7.17	723
10	7.17	723
Media	6.59	805.67

Tabla 20 : SCP en virtual y local, con hip

4.2.1.2 - Entorno real

Intento	Tiempo de descarga(min)	Velocidad media(KB/s)
1	4.08	911.5
2	4.01	938
3	3.34	1000
4	4.05	922.7
5	3.51	978.6
6	3.36	1000
7	4.07	915.2
8	3.37	1000
9	4.14	890
10	4.06	918.8
Media:	3.8	947.48

Tabla 21 : SCP en real y local, con hip

Conclusiones

Al comparar los resultados de estas pruebas vemos que al transferir un fichero, en las máquinas virtuales, suele tardar un poco más y no alcanza la velocidad máxima en la mayoría de los casos que sería entorno a mega por segundo. En cambio, en el entorno real ha ido excepto un par de veces, a su máxima velocidad (la que la tarjeta de red le permitía) Las velocidades de transferencia han variado muy poco en la mayoría de los casos menos en el caso del entorno virtual que se han registrado hasta 623 KB/s frente a los 1000 de máxima. En este caso el entorno real ha sido mucho más regular.

4.2.2 - Transferencia en distinta red

Descripción

La segunda prueba realizada con el comando scp, como bien se ha comentado con anterioridad, ha consistido en realizar una transferencia de un fichero de 221 MB desde un ordenador situado en la red 56.0 y otra en la 57.0. En este segundo caso, el principal objetivo es comparar los tiempos obtenidos en las anteriores pruebas para estudiar los posibles retardos introducidos por el protocolo HIP al encontrarse en otra red.

Como en los anteriores casos, las pruebas se han realizado en entorno virtual y real.

Resultados obtenidos

En las siguientes tablas se muestran los tiempos de descarga y la velocidad de transferencia obtenidos en la situación descrita anteriormente. La primera tabla corresponde al entorno virtualizado y la segunda al real.

4.2.2.1 - Entorno virtual

Intento	Tiempo de descarga(s)	Velocidad media(KB/s)
1	6.05	619.3
2	4.16	883
3	5.53	640.4
4	5.52	642.2
5	4	941.9
6	4.21	866.1
7	5.49	647.7
8	5.36	672.8
9	5.38	668.8
10	5.43	659.1
Media:	5.11	724.13

Tabla 22 : SCP en virtual y pública, con hip

4.2.2.2 - Entorno real

Intento	Tiempo de descarga(min)	Velocidad media(KB/s)
1	9.26	399.4
2	12.05	311.8
3	10.35	356
4	15.45	239.2
5	24.12	155.7
6	23.26	160.8
7	5.54	638.6
8	10.41	352.7
9	13.55	270.7
10	13.27	280.1
Media:	13.76	316.5

Tabla 23 : SCP en real y pública, con hip

Conclusiones

Esta vez se puede ver que la velocidad ha sido mucho menor, y por tanto, el tiempo de descarga mayor, en la infraestructura real. La velocidad en el entorno virtual ha sido muy parecida a las pruebas anteriores, las pruebas donde el cliente se encontraba en la misma red que el servidor.

Las velocidades en el entorno real han variado desde 155.7 que ha sido la más pequeña hasta 638.6, siendo estos dos picos los únicos, puesto que las demás velocidades han rondado entorno a los 300KB/s. Aquí se puede apreciar que la velocidad disminuye mucho con HIP, cuando el servidor se encuentra en otra red y por tanto se debe de hacer uso del protocolo.

4.2.3 - Cambio de red durante la transferencia

Descripción

Para finalizar con el comando scp, se ha efectuado un tipo de pruebas en las que se ha efectuado un cambio de red del cliente. El cliente accede al servidor y copia un archivo. El objetivo principal de esta prueba ha sido el de medir el tiempo que se tarda en realizar el cambio de red, es decir, el tiempo que se tarda desde que, estando la transferencia en curso, se inicia el proceso de cambio de red por parte del emisor, hasta que la transferencia vuelve a estar completamente operativa. Este proceso requiere ejecutar el script de cambio de red, una vez se haya iniciado la transferencia.

Resultados obtenidos

En las siguientes tablas se muestran los tiempos necesitados por el sistema para hacer el cambio del cliente. La primera tabla especifica los resultados obtenidos con la infraestructura virtual, y la segunda con las máquinas reales.

4.2.3.1 - Maquina virtual

Intento	Tiempo de cambio(s)
1	18.1
2	14.3
3	20.5
4	32
5	18.3
6	20.4
7	19.9
8	17.6
9	16.6
10	17.9
Media:	19.56

Tabla 24 : Tiempos de reconexión de SCP en virtual con hip

4.2.3.2 - Máquina real

Intento	Tiempo de cambio(s)
1	32.2
2	31.6
3	33.6
4	28.8
5	60.5
6	30
7	34
8	33.5
9	28.6
10	32.5
Media:	34.53

Tabla 25 : Tiempos de reconexión de SCP en real con hip

Conclusiones

El tiempo que tarda la conexión en restablecerse es muy parecida en las 10 pruebas que se han realizado, sea bien en máquinas virtuales como en la infraestructura real. Los tiempos para las máquinas virtuales rondan los 19.56 segundos menos una vez que el tiempo se ha disparado hasta los 32 segundos, debido al demonio Hip y el fallo de autenticación del cliente al cambiarse de red.

Lo mismo pasa en la infraestructura real, solo que en este caso los tiempo son mayores, prácticamente el doble. El pico sube hasta el minuto de espera y la media del tiempo de reconexión es de 34.53 segundos.

5.- FTP

Descripción de la prueba

Para realizar el estudio de la transferencia de un fichero a través del protocolo ftp, se realizó estando el servidor y el cliente en la misma red primero, y luego se repitió la prueba estando el servidor en una red diferente. El motivo de no hacer el cambio durante la transferencia del fichero fue que ftp no muestra información de la descarga mientras esta dura, por lo que no se podría saber el tiempo de reconexión, que es el dato que se quiere obtener con dicha prueba.

Se adjuntan las imágenes de las dos situaciones sobre el que se realiza la prueba. Obviamente, y como ya se ha explicado anteriormente, y realizado para las pruebas anteriores, las pruebas se realizaran primero en estructura virtual y luego en la estructura real, y sobre ambos protocolos de movilidad (hip y openvpn).

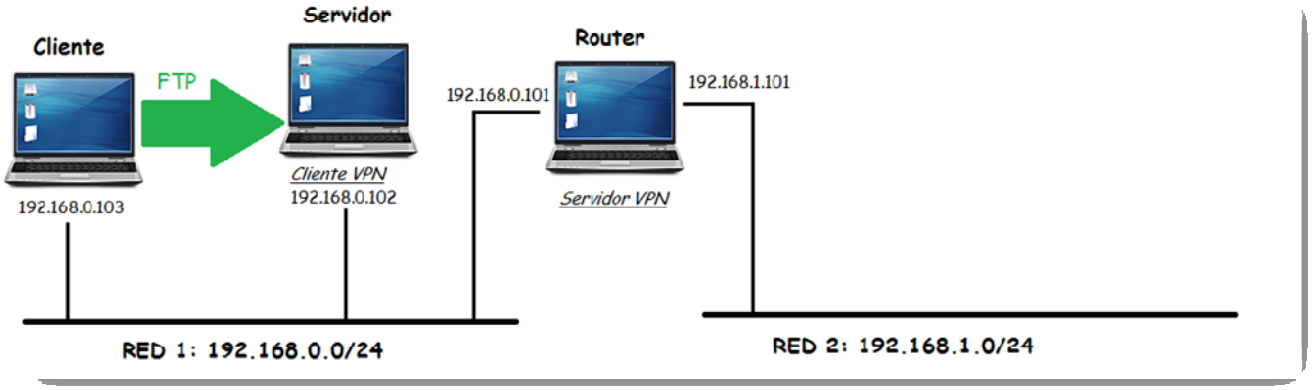


Ilustración 31 : FTP en red local

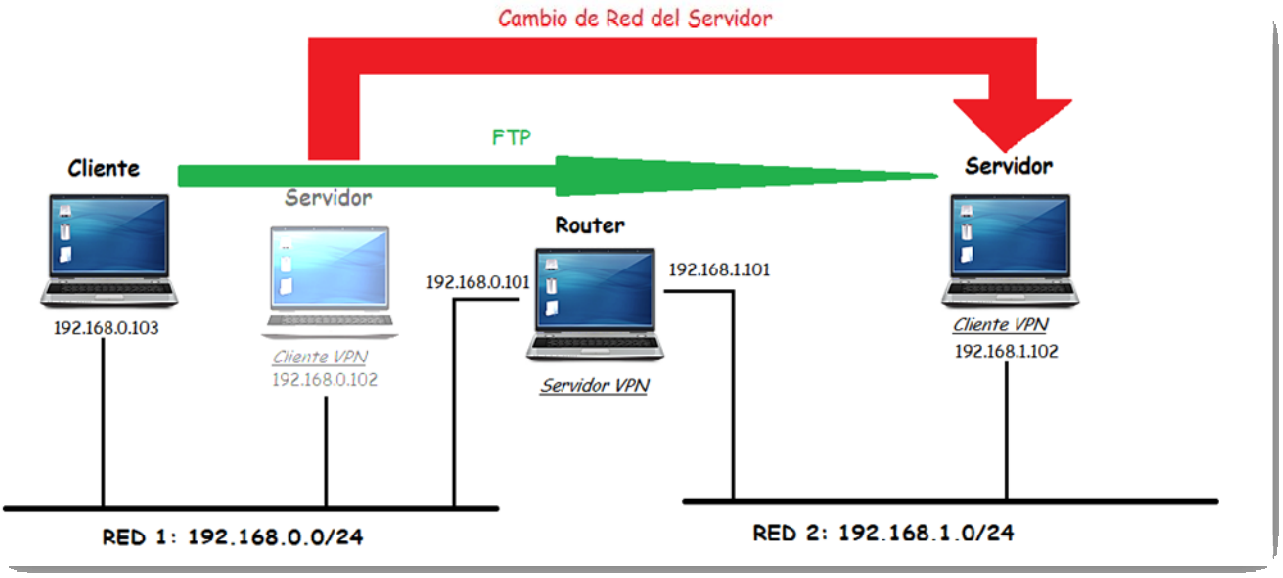


Ilustración 32 : FTP en red pública

El objetivo de la prueba era estudiar el estado de la red durante la descarga de un fichero a través del protocolo ftp, y ver la diferencia entre descargarlo a través de una red local y a través de una red con openvpn.

Ftp es un protocolo de red utilizado para la transferencia de archivos entre sistemas conectados a una red TCP, y que está basado en una arquitectura cliente-servidor. El cliente se conecta contra el servidor, y a partir de ahí, puede descargarse ficheros que estén en el servidor, o depositar archivos suyos en el servidor, a partir de unos comandos. El puerto que suele utilizar el protocolo ftp es el 20 o el 21. Es un protocolo que proporciona una gran velocidad en la descarga de ficheros, pero tiene un problema, y es la seguridad, pues todo se efectúa en texto plano, y es fácil que pueda haber problemas de seguridad.

Para el desarrollo de esta prueba se utilizaron tres ficheros de diversos tamaños, para ver el tiempo que se tardaba en la descarga de los mismos. Estos ficheros fueron:

- *pequeFTP*: fichero de texto de un tamaño de 32.4 MB.
- *medioFTP*: fichero de texto de un tamaño de 308 MB.
- *grandeFTP*: fichero de texto de un tamaño de 3.01 GB.

Para ejecutar la prueba, se utilizó el comando get, que sirve para coger un fichero del servidor y colocarlo en el cliente. Aquí se adjunta las instrucciones que se ejecutaron para conseguir la descarga del fichero:

ftp direccionIP (dirección local del servidor ftp).

get -DireccionFicheroFuente –DireccionFicheroDestino (copia el fichero fuente en la dirección que se le indique).

Se va a realizar el estudio de los dos protocolos de los que consta la memoria: VPN y HIP.

5.1 - VPN

La tecnología Vpn había que estudiarla en el entorno de las máquinas virtuales, y en la estructura real. Además, era necesaria la existencia de unos datos de referencia, por lo que se procedió a la descarga de ficheros en una red local, y después a través de Openvpn, para poder comparar los resultados.

5.1.1 - Virtual

Inicialmente la idea era descargar los tres ficheros desde todas las opciones posibles enunciadas anteriormente, pero por razones que ahora se describirán en cada caso particular, no fue posible, y se descargó el fichero que mejor se amoldaba a la red en cada momento.

5.1.1.1 - Red Local

Se procedió primero a la descarga del fichero *pequeFTP* (32.4 MB) en la red local, pero al ejecutarlo dos veces, los resultados obtenidos eran mínimos (menores de 2 segundos), por lo que se decidió no hacer análisis de un fichero que daba unos tiempos tan pequeños.

Después, se descargó medioFTP (308 MB) en la red local, y se almacenaron los datos de la descarga para poder estudiarlos. Aquí se adjunta la tabla con los datos de las descargas. Ésta se hizo 10 veces para tener un buen número de muestras, y que la prueba tuviera mayor fiabilidad.

DESCARGA DEL FICHERO MEDIOFTP		
Intento	**Tiempo de descarga (s)**	**Velocidad media (kB/s)**
1	14.73	21444.6
2	18.22	17341.1
3	17.58	17976.2
4	16.63	19000.2
5	20.16	15672.1
6	23.22	13604.2
7	21.30	14830.6
8	17.80	17749.1
9	16.83	18774.2
10	14.42	21903.8
Media	18.09	17829.6

Tabla 26 : FTP en virtual y local, con vpn, de medioFTP

A continuación, se descargó *grandeFTP* (3.01 GB) también en la red local, para ver el comportamiento de los pc´s con dos ficheros de distintos tamaños. Al igual que en el caso anterior, se adjunta la tabla con las 10 pruebas que se ejecutaron sobre el fichero.

DESCARGA DEL FICHERO MEDIOFTP		
Intento	**Tiempo de descarga (s)**	**Velocidad media (kB/s)**
1	157.67	20038.4
2	187.63	16838.3
3	215.70	14647.2
4	233.38	13537.9
5	200.88	15728.0
6	222.76	14183.1
7	195.87	16130.5
8	184.03	17168.1
9	170.42	18539.2
10	166.70	18952.4
Media	193.50	16576.3

Tabla 27 : FTP en virtual y local, con vpn de grandeFTP

5.1.1.2 - A través de OpenVPN

Una vez que se conectaban el servidor ftp y el cliente ftp en redes distintas, y se iniciaba el servicio OpenVPN (el servidor ftp era cliente vpn), el servidor ftp adquiría una dirección IP de la red local, que se la otorgaba el servidor openvpn, por lo que, virtualmente, estaban en la misma red. Al ejecutar ftp sobre la dirección de la red local que se le había asignado al servidor ftp, se realizaba la conexión entre el servidor ftp y el cliente ftp, y se podía proceder a la descarga del fichero. Inicialmente, se descargó el fichero *pequeFTP* (32.4 MB), para comprobar si el tiempo era similar al de la red local y no merecía la pena hacer su estudio, o si por el contrario a través de openvpn el tiempo se disparaba, y sí que era interesante hacer su estudio.

Efectivamente, se dio este segundo caso, pues los tiempos eran mucho mayores para descargar el fichero pequeño a través de openvpn que el tiempo de descarga del fichero grande en la red local, como se puede apreciar en la tabla.

DESCARGA DEL FICHERO PEQUEFTP		
Intento	**Tiempo de descarga (s)**	**Velocidad media (kB/s)**
1	306.37	108.6
2	303.08	109.7
3	241.06	138.0
4	243.71	136.5
5	244.70	135.9
6	230.66	144.2
7	241.83	137.5
8	253.37	131.3
9	230.33	144.4

10	229.04	145.2
Media	252.42	133.1

Tabla 28 : FTP en virtual y pública, con vpn

Dado que los tiempos para este fichero ya eran muy grandes, se decidió no hacer las pruebas para los ficheros *medioFTP* (308 MB) y *grandeFTP* (3.01 GB), pues si con un fichero tan pequeño los tiempos eran grandes, con el resto de ficheros se dispararían, y no tiene ningún sentido compararlo con nada de local.

Conclusiones

La red local en las máquinas virtuales da una velocidad de red de casi 20MB/s, lo cual es una velocidad muy alta. La velocidad media de descarga con openvpn a través de las máquinas virtuales ha sido de 133.1 KB/s, por lo que queda claro que es una diferencia abismal hacerlo entre red local o a través de openvpn. En máquinas virtuales, openvpn ralentiza muchísimo las descargas de ficheros a través de ftp, algo más de 130 veces más lenta que trabajar en la red local.

5.1.2 - Real

Una vez estudiada la descarga de ficheros ftp, y descartados los ficheros medioFTP y grandeFTP por ser demasiado grandes para los resultados que se han obteniendo, se procedió a ejecutar la misma prueba, pero esta vez en la estructura real, para ver su comportamiento teniendo máquinas físicos y enrutadores físicos, por los que tener que enviar los datos. Al igual que antes, se realizó primero la prueba en la red local, y después el servidor ftp y el cliente ftp en redes separadas, y con el servicio openvpn en marcha.

5.1.2.1 - Red Local

Conectados el servidor ftp y el cliente ftp a la misma red local, se procedió a la descarga del fichero *pequeFTP* (32.4 MB), para ver la velocidad y el tiempo de descarga en una red local física, y tener una referencia para luego comparar con la prueba de openvpn. Aquí se adjuntan los datos de los 10 intentos que se hicieron de la prueba.

DESCARGA DEL FICHERO PEQUEFTP		
Intento	**Tiempo de descarga (s)**	**Velocidad media (kB/s)**
1	32.68	1017.8
2	30.90	1076.2
3	30.61	1086.6
4	33.14	1003.5
5	30.59	1087.2
6	28.94	1149.2
7	28.73	1157.7
8	29.78	1116.7
9	29.53	1126.1

10	30.19	1101.7
Media	30.51	1092.3

Tabla 29 : FTP en real y local, con vpn

5.1.2.2 - A través de OpenVpn

Una vez hecha la prueba en local, se pasó el cliente openvpn a otra red, y se inició el servicio openvpn para que le correspondiese una dirección ip de la red local, en la que se encontraba el cliente ftp, y se inició la descarga a través de esa dirección ip. Una vez conectados servidor y cliente, se produjo a la descarga del fichero *pequeFTP* (32.4 MB) 10 veces, para después sacar los valores medios y poder comparar.

DESCARGA DEL FICHERO PEQUEFTP		
Intento	**Tiempo de descarga (s)**	**Velocidad media (kB/s)**
1	1497.51	22.2
2	1690.33	19.7
3	1638.28	20.3
4	1705.49	19.5
5	1504.85	22.1
6	1614.42	20.6
7	1546.84	21.5
8	1591.25	20.9
9	1554.07	21.4
10	1671.21	19.9
Media	1601.43	20.8

Tabla 30 : FTP en real y pública, con vpn

El tiempo medio de descarga de *pequeFTP* ha sido de 16014.25 segundos, lo que equivale a más de 26 minutos.

Conclusiones

Al igual que pasaba en las máquinas virtuales, el tiempo a través de openvpn es mucho mayor que en la red local, pero ahora la diferencia no es tan grande, pues la velocidad baja de 1092.3 kB/s a 20.8 kB/s, lo que supone que la velocidad en local es 52,5 veces mayor. 20.8 kB/s no es una velocidad demasiado alta como para asegurar que openvpn tiene un buen comportamiento en la transmisión de ficheros ftp.

5.2 - HIP

Al igual que en openvpn, inicialmente se van a realizar las descargas en máquinas virtuales, y después se procederá a descargar el fichero en la estructura real, para tener un doble punto de vista del comportamiento de HIP con la descarga de ficheros mediante el protocolo ftp.

5.2.1 - Virtual

Para comparar los resultados de la descarga del fichero con el protocolo ftp a través de HIP, inicialmente se descarga el fichero en la red local, y después teniendo el servidor ftp en otra red, y descargando a partir de la dirección que otorgará hip.

5.2.1.1 - Red Local

Esta prueba es la misma que ya se hizo en el apartado anterior, para openvpn, por lo que no hay más que decir sobre ella. Para consultarla, observar la tabla de ftp para descarga en local en máquinas virtuales, en openvpn.

5.2.1.2 - A través de HIP

Una vez instaurado el servidor ftp y el cliente ftp en redes distintas, y teniendo HIP funcionando en ambos dos, se procede a la descarga del fichero pequeFTP (32.4 MB), para poder estudiar sus tiempos. Como en las demás pruebas, se descargará el fichero 10 veces para que el resultado de la prueba sea más fiable.

DESCARGA DEL FICHERO PEQUEFTP		
Intento	**Tiempo de descarga (s)**	**Velocidad media (kB/s)**
1	18.47	1800.3
2	19.43	1711.5
3	18.09	1838.2
4	18.04	1843.6
5	18.44	1803.6
6	19.14	1737.7
7	18.22	1825.3
8	18.65	1783.6
9	18.08	1839.8
10	20.28	1640.2
Media	18.68	1782.4

Tabla 31 : FTP en virtual y pública, con hip

Conclusiones

Si se comparan las velocidades medias de trabajar en red local o hacerlo a través de HIP en las máquinas virtuales, se puede ver que la velocidad media varía de 17829.6 kB/s en la red local a 1782.4 kB/s, lo cual indica que la velocidad en un sitio es 10 veces mayor que en el otro aproximadamente. Es una diferencia grande, pero mucho menor que la que se obtenía con openvpn, en la que la velocidad media era de 133.1 kB/s, más de 13 veces más lento que en HIP.

5.2.2 - Real

Por último, y al igual que se hizo en openvpn, se repiten las pruebas en la estructura real, con los pc´s y routers indicados anteriormente. Primero se hará la prueba en la red local de la estructura real para tener un punto sobre el que comparar, y después se descargará el fichero a través de HIP, para así poder ver las diferencias de tiempos y de velocidades de descargas cuando se activa HIP y cuando no.

5.2.2.1 - Red Local

Esta prueba es la misma que ya se hizo en el apartado anterior, para openvpn, por lo que no hay más que decir sobre ella. Para consultarla, observar la tabla de ftp para descarga en local en estructura real, en openvpn.

5.2.2.2 - A través de HIP

Como en las pruebas anteriores, se procede a descargar pequeFTP (32.4 MB) desde el cliente ftp, cogiéndolo en el servidor ftp, a través de HIP, estando cada uno de ellos en una red distinta y con HIP activado. Se inicia la conexión a partir de la dirección que nos proporciona HIP, y se repite la descarga 10 veces, de las cuales se adjuntan aquí a continuación los resultados:

DESCARGA DEL FICHERO PEQUEFTP		
Intento	**Tiempo de descarga (s)**	**Velocidad media (kB/s)**
1	122.94	270.5
2	102.30	325.1
3	63.85	520.9
4	81.62	407.5
5	175.53	189.5
6	53.89	617.1
7	56.24	591.4
8	54.24	613.2
9	53.05	626.9
10	60.07	553.6
Media	82.37	471.6

Tabla 32 : FTP en real y pública, con hip

Conclusiones

Ahora, el tiempo medio de descarga ha sido de 471.6 kB/s, por los 1092.3 kB/s que se tenía en la red local en la estructura real. En comparación a las máquinas virtuales, los relación de los tiempos entre los de la red local y los obtenidos a través de HIP, en la estructura real son mucho mejores, pues son solo 2.3 veces peores, por las 10 que lo eran en las máquinas virtuales. En comparación a openvpn, se obtienen unos resultados mucho mejores, pues a través de openvpn la velocidad era sólo de 20.8 kB/s.

6.- HPING

Por último, se adjuntan los resultados de las pruebas que se hicieron para estudiar hping sobre los protocolos. Se ejecutaron dos pruebas sobre ambos protocolos: primero el escaneo de puertos en la red local, y después el escaneo cuando el servidor ha cambiado de red. Estas situaciones quedan reflejadas en las dos siguientes ilustraciones:

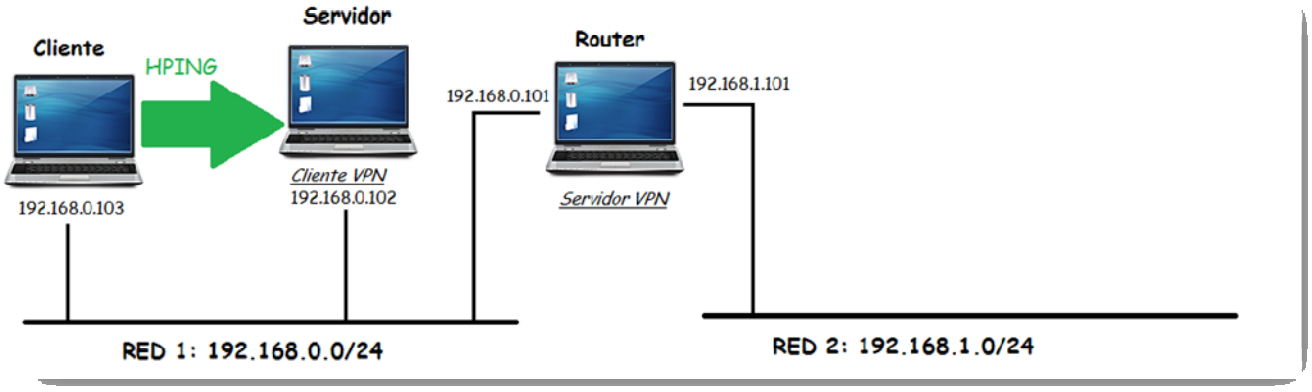


Ilustración 33 : HPING en red local

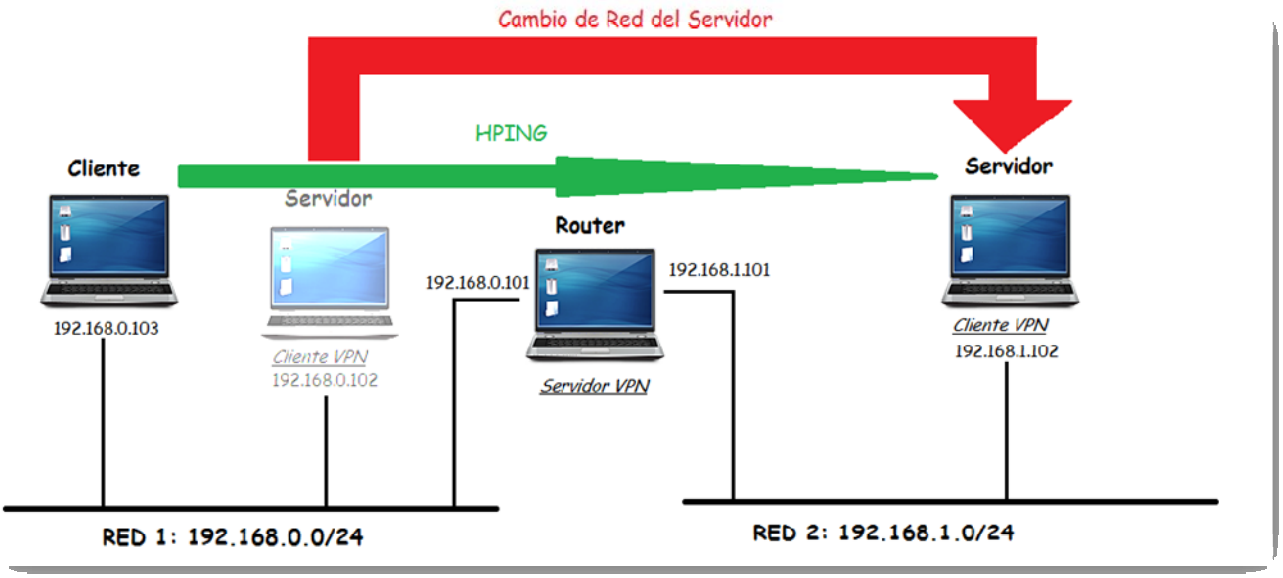


Ilustración 34 : HPING en red pública

6.1 - VPN

Descripción de las pruebas

Las pruebas con hping han consistido en hacer un escaneo de puertos de una máquina remota. Este escaneo de puertos comprende el rango entre el puerto 1 y el puerto 150. La herramienta hping3 lleva a cabo el escaneo de puerto mediante el envío de paquetes TCP con el flag SYN activado, para esperar después la respuesta desde la máquina remota. En el caso de que el puerto este abierto la respuesta contendrá los flags SYN, ACK, y en caso contrario RST, ACK. El objetivo de estas pruebas, una vez más ha sido ver la diferencia de tiempos existentes cuando la comunicación se da en la misma red y cuando esta sucede a través de OpenVPN.

Resultados obtenidos

En las siguientes figuras se muestran los tiempos de respuesta que se han medido al hacer las peticiones a los primeros 150 puertos de la máquina remota. Todas estas gráficas corresponden a los intentos realizados cuando ambos ordenadores se encuentran en la misma red.

Número de peticiones: 153
Número de respuestas: 150
Tiempo medio de respuesta: 0.0011401622

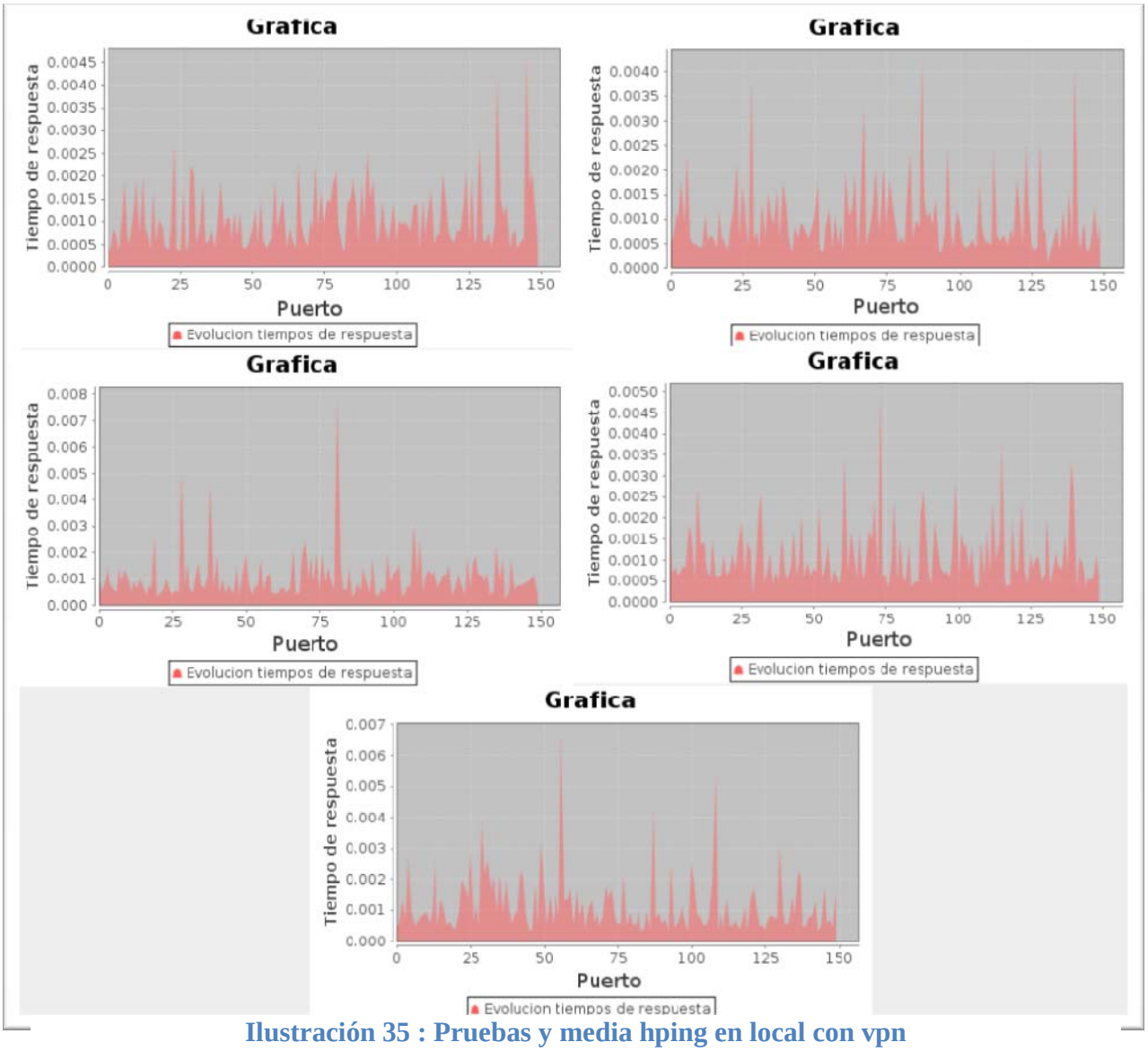


Ilustración 35 : Pruebas y media hping en local con vpn

En las siguientes figuras se muestran los tiempos de respuesta que se han medido al hacer las peticiones a los primeros 150 puertos de la máquina remota. Todas estas gráficas corresponden a los intentos realizados cuando los ordenadores se encuentran en una red diferente, y enviando el tráfico a través de VPN.

Número de peticiones: 153
Número de respuestas: 150
Tiempo medio de respuesta: 0.0038018445

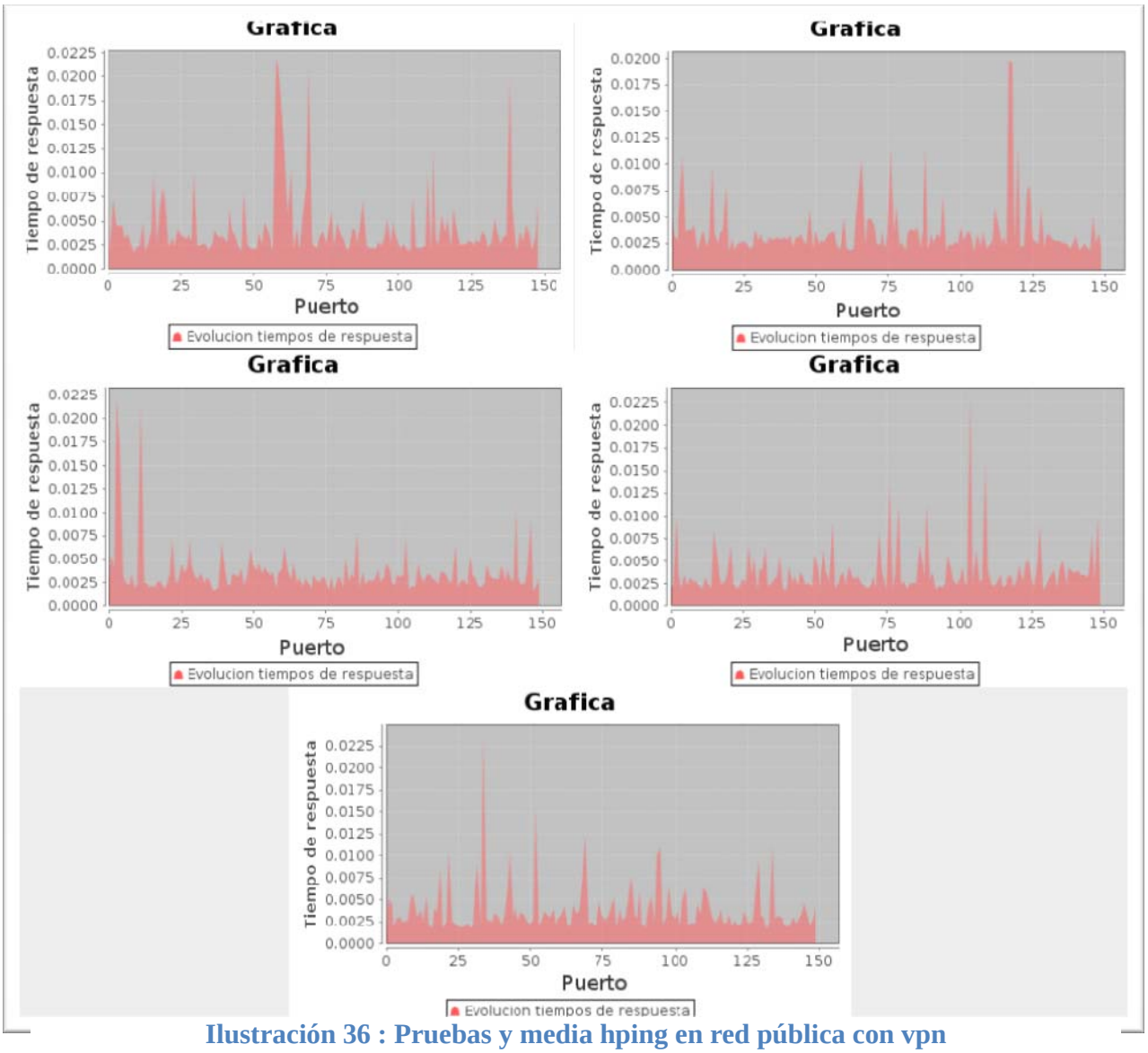


Ilustración 36 : Pruebas y media hping en red pública con vpn

Conclusiones

Los resultados, nuevamente, han sido los esperados. Se han obtenido tiempos de respuesta más altos al utilizar VPN que al estar en la misma red. Sin embargo, a pesar de que la respuesta tarda más en llegar y que esta diferencia no es para nada despreciable, el funcionamiento de la herramienta no se ve perjudicado de forma crítica por el uso de VPN, y para un uso que no tenga el factor tiempo como crítico, esta demora no es para nada importante.

6.2 - HIP

Descripción de las pruebas

Las mismas pruebas se han realizado en HIP. Se han realizado 10 pruebas: 5 de ellas en la red 192.168.56.0 y otras 5 en la 192.168.57.0. Los dos en entorno virtual.

Resultados obtenidos

En las siguientes figuras se muestran los tiempos de respuesta que se han medido al hacer las peticiones a los primeros 152 puertos de la máquina remota. Todas estas gráficas corresponden a los intentos realizados cuando ambos ordenadores se encuentran en la misma red.

Número de peticiones: 152
Número de respuestas: 150
Tiempo medio de respuesta: 0.005441093

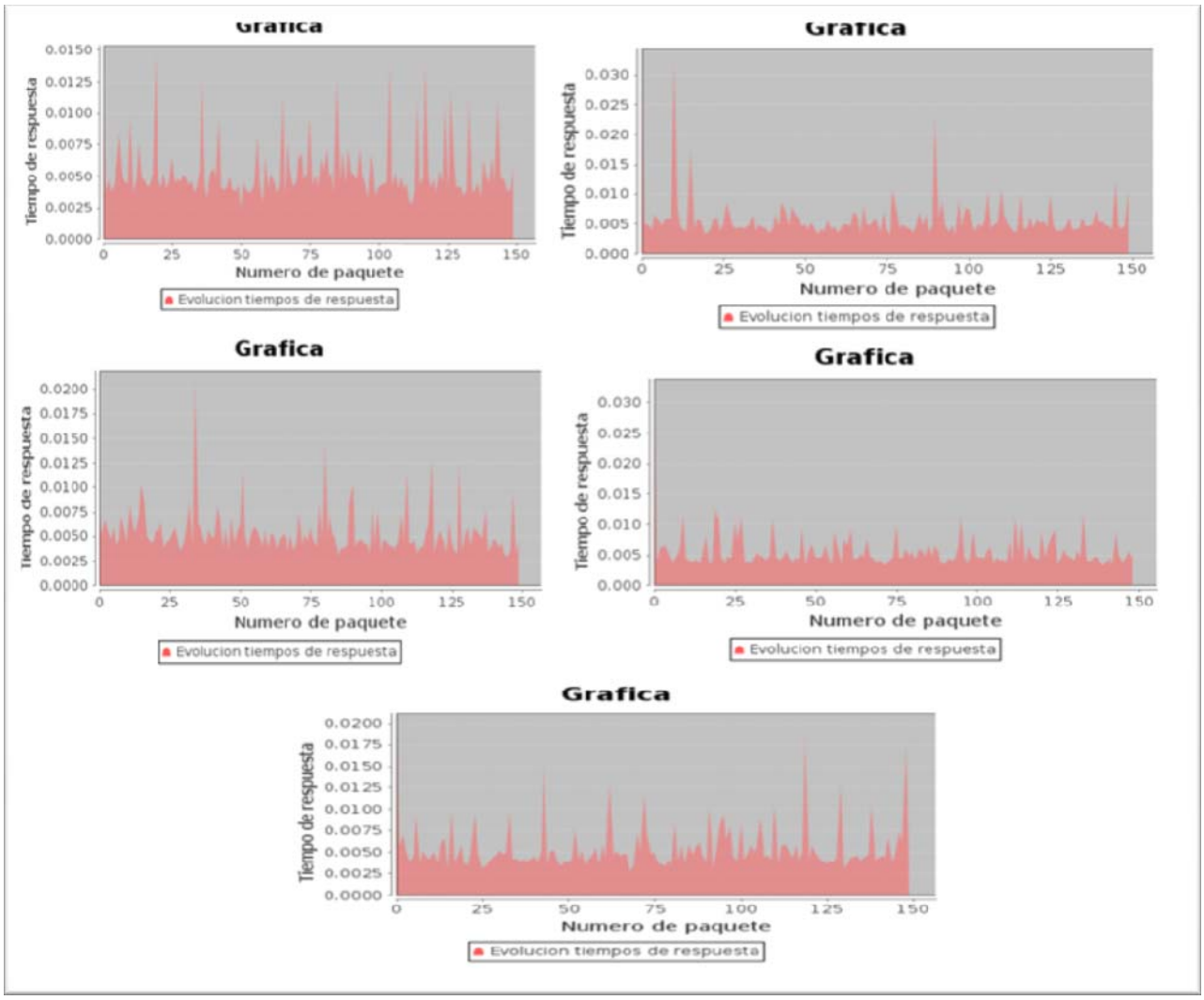


Ilustración 37 : Pruebas y media hping en local con hip

En las siguientes figuras se muestran los tiempos de respuesta que se han medido al hacer las peticiones a los primeros 150 puertos de la máquina remota. Todas estas gráficas corresponden a los intentos realizados cuando los ordenadores se encuentran en una red diferente, y enviando el tráfico con la tecnología HIP.

Number of requests: 152
Number of responses: 150
Average response time: 0.0071480465

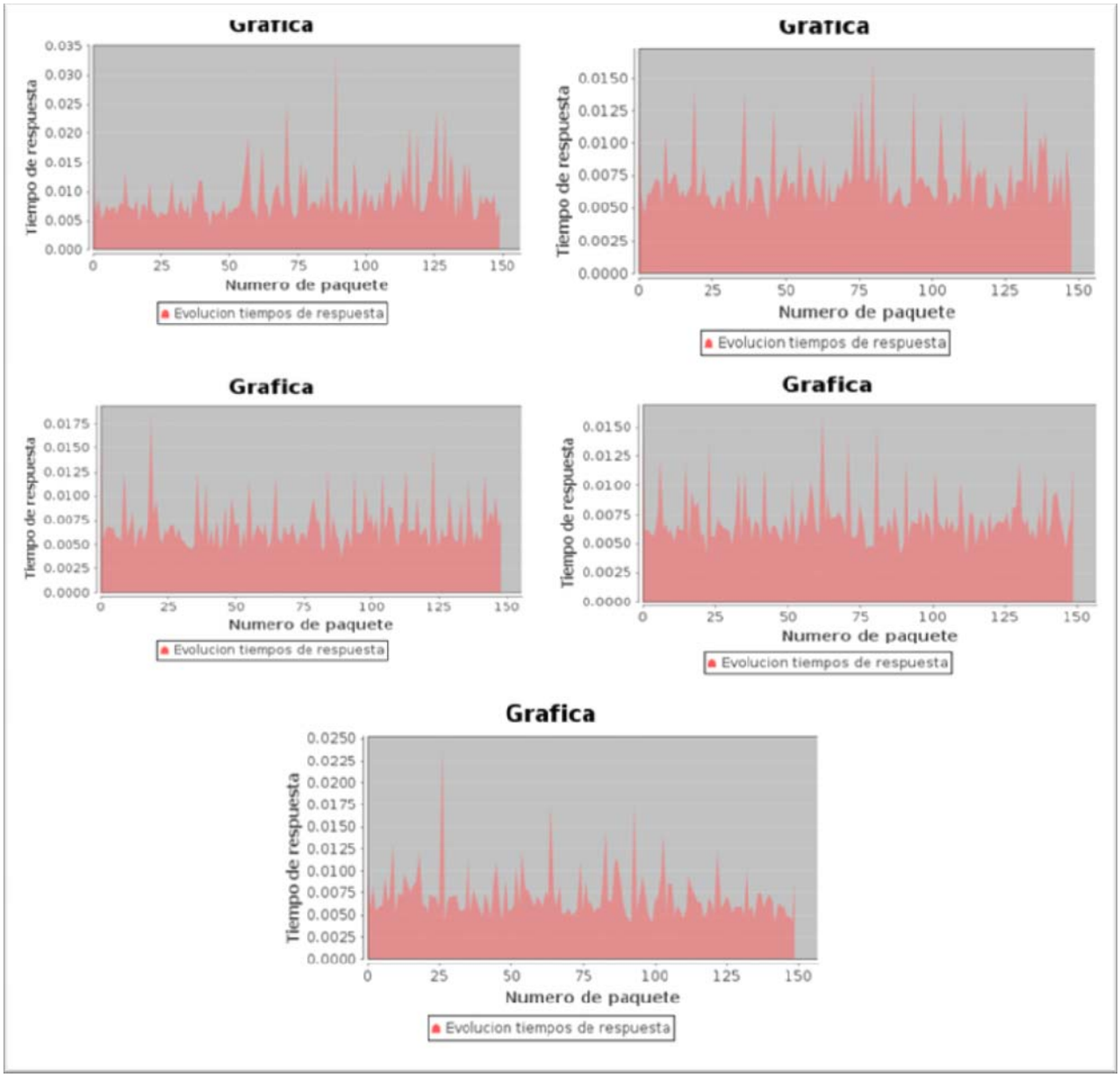


Ilustración 38 : Pruebas y media hping en red pública con hip

Conclusiones

Los resultados no sorprenden. El tiempo de respuesta cuando el cliente estaba en la red 57.0 es ligeramente mayor que cuando el cliente se encontraba en la misma red que el servidor. Los datos obtenidos de las gráficas muestran que las peticiones no difieren mucho los unos de los otros, habiendo una diferencia de 0.005 segundos para la infraestructura virtual y 0.01 segundos para la infraestructura real.

Comparación general

En las siguientes páginas se muestran algunas tablas resumen en las que se intenta mostrar de forma clara las diferencias entre las dos tecnologías. Dado que éste es un proyecto de comparación de dos tecnologías, se han confeccionado estas tablas para dar una visión global de ambas para que pueda verse una comparación entre las tecnologías sin tener que entrar en la maraña de números que son las pruebas especificadas en la anterior sección de la memoria.

Las características que se han evaluado en estas tablas son las siguientes:

- **Altera la pila de protocolos**: Es un valor Sí/No dependiendo de si la tecnología estudiada introduce algún cambio en la pila de protocolos, es decir, si añade o modifica capas en el modelo OSI.
- **Tiempo de reconfiguración**: Especifica el tiempo que se tarda en transportar el servidor de una red a otra y que este comience a estar completamente operativo haciendo uso de la tecnología a evaluar.
- **Tiempo de más necesitado**: Especifica el tiempo de más que se necesita para hacer una transferencia de un ordenador a otro usando una de las tecnologías evaluadas, comparándolo con el tiempo necesario para esa misma transferencia estando ambos computadores en la misma red y comunicándose directamente. Este valor está especificado en valores proporcionales. Por ejemplo, si el valor "Tiempo de más necesitado" es de 2 para una tecnología dada, quiere decir que usando esa tecnología hace falta el doble de tiempo para hacer una transferencia comparando con el tiempo necesario en el caso en el que los ordenadores están conectados directamente.

Comparativa general en entorno virtual

		HIP	VPN
PING	Altera la pila de protocolo	Si	No
	Tiempo de reconfiguración	12.1s	5.2s
	Tiempo de más necesitado	1.13	4.65
PETICIONES WEB	Tiempo de reconfiguración	16.16s	7.5
	Tiempo de más necesitado	1.05	1.73
DESCARGA HTTP	Tiempo de reconfiguración	16.16s	7.47s
	Tiempo de más necesitado	2.11	54.1
SCP	Tiempo de reconfiguración	19.56s	13s
	Tiempo de más necesitado	-1.29*	66.9
FTP	Tiempo de reconfiguración	No medido	No medido
	Tiempo de más necesitado	9.75	9.5
HPING	Tiempo de reconfiguración	No medido	No medido
	Tiempo de más necesitado	1.31	3.33

Tabla 33 : Comparativa general en entorno virtual de HIP & OpenVPN

*En las mediciones realizadas, las pruebas en red local resultaron más lentas que las realizadas a través de HIP. Estos resultados no son los esperados, pero a pesar de esto se ha rotulado de verde porque el tiempo necesitado por HIP ha sido menor que VPN.

Comparativa general en entorno real

		HIP	VPN
	Altera la pila de protocolo	Sí	No
PING	Tiempo de reconfiguración	22.5s	29.1s
	Tiempo de más necesitado	1.77	2.37
PETICIONES WEB	Tiempo de reconfiguración	20s	40s
	Tiempo de más necesitado	1.06	2.75
DESCARGA HTTP	Tiempo de reconfiguración	31.69s	19.42s
	Tiempo de más necesitado	1.93	21.29
SCP	Tiempo de reconfiguración	34.53s	1m 49s
	Tiempo de más necesitado	3.62	18.67
FTP	Tiempo de reconfiguración	No medido	No medido
	Tiempo de más necesitado	2.63	52.49

Tabla 34 : Comparativa general en entorno real de HIP & OpenVPN

GRAFICAS COMPARATIVAS

Ping

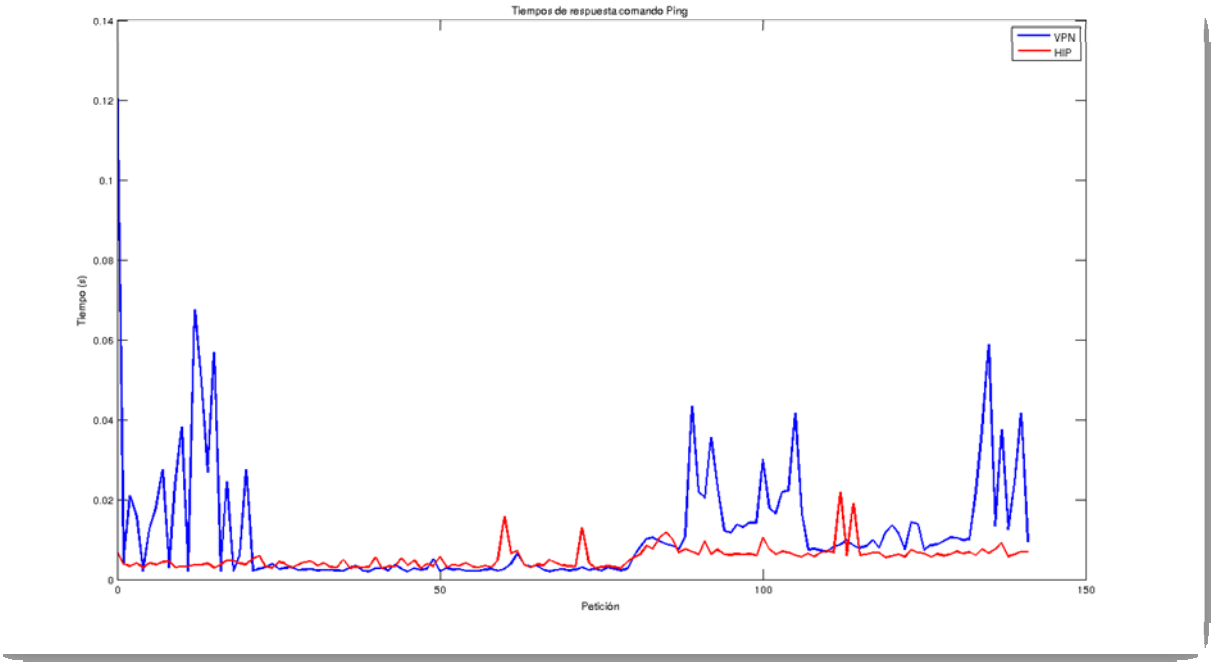


Ilustración 39 : Comparativa con Ping entre OpenVPN & HIP

En esta primera gráfica comparativa, aparecen los tiempos medios de las pruebas de ping en el entorno real. Como puede observarse en ambos casos los tiempos de respuesta son parecidos antes de efectuarse el cambio a la red pública (exceptuando algunos picos en el caso de OpenVPN). Una vez efectuado el cambio (aproximadamente a partir de la petición 80), se ve como OpenVPN tiende a necesitar más tiempo que HIP.

Peticiones HTTP

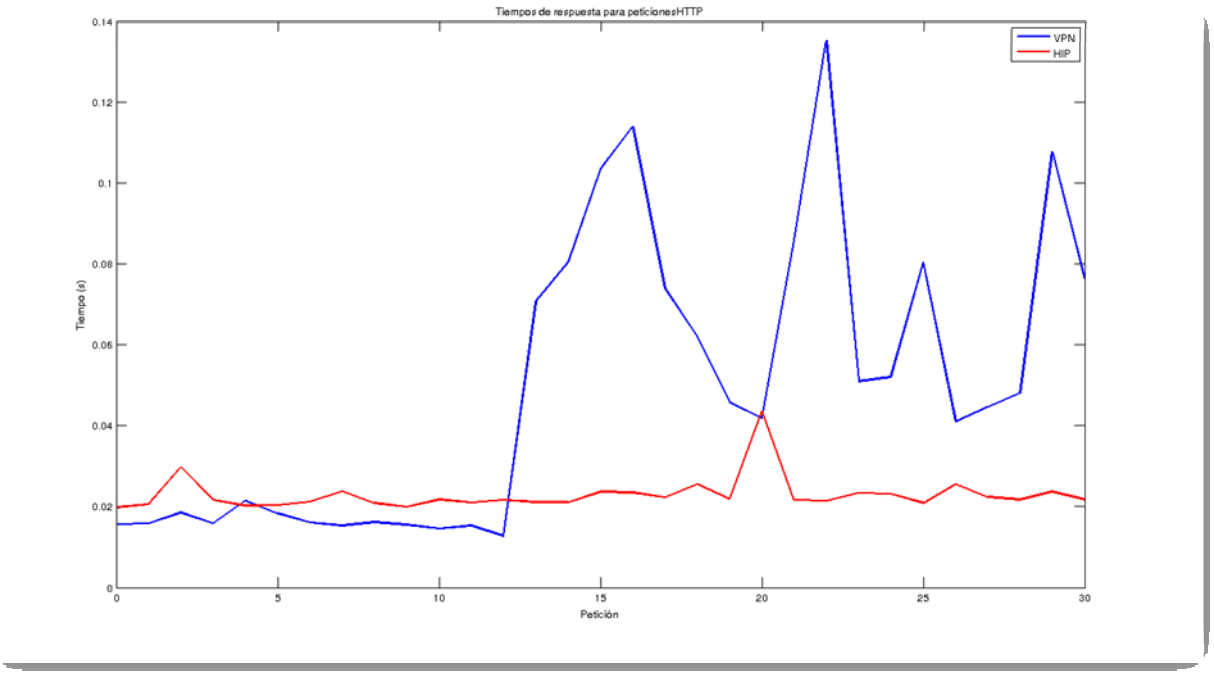


Ilustración 40 : Comparativa con peticiones http entre OpenVPN & HIP

En este segundo caso, se observa un comportamiento similar. A partir de la petición 16 el tráfico se efectúa a través de HIP y OpenVPN. Puede verse claramente que OpenVPN tiene un efecto negativo en la velocidad de transferencia, mientras que el tiempo con HIP permanece prácticamente inalterado.

Descarga de fichero mediante HTTP

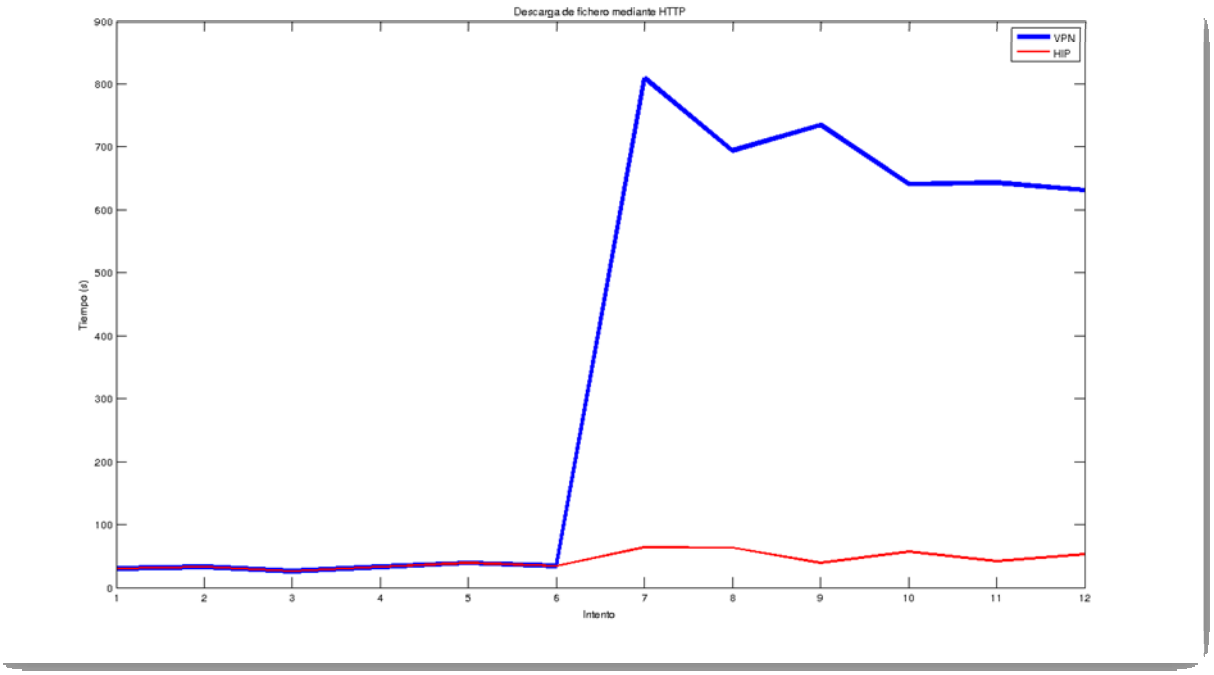


Ilustración 41 : Comparativa con descarga mediante http entre OpenVPN & HIP

En este gráfico, en el que nuevamente se muestra una transferencia HTTP (siendo esta vez una transferencia de un fichero más grande que en el anterior gráfico) el comportamiento de ambas tecnologías se repite. Hasta el sexto intento ambos tienen un tiempo equivalente y a partir de ahí OpenVPN refleja un aumento en el tiempo de respuesta mientras que HIP no produce un efecto demasiado apreciable.

Descarga de fichero mediante scp

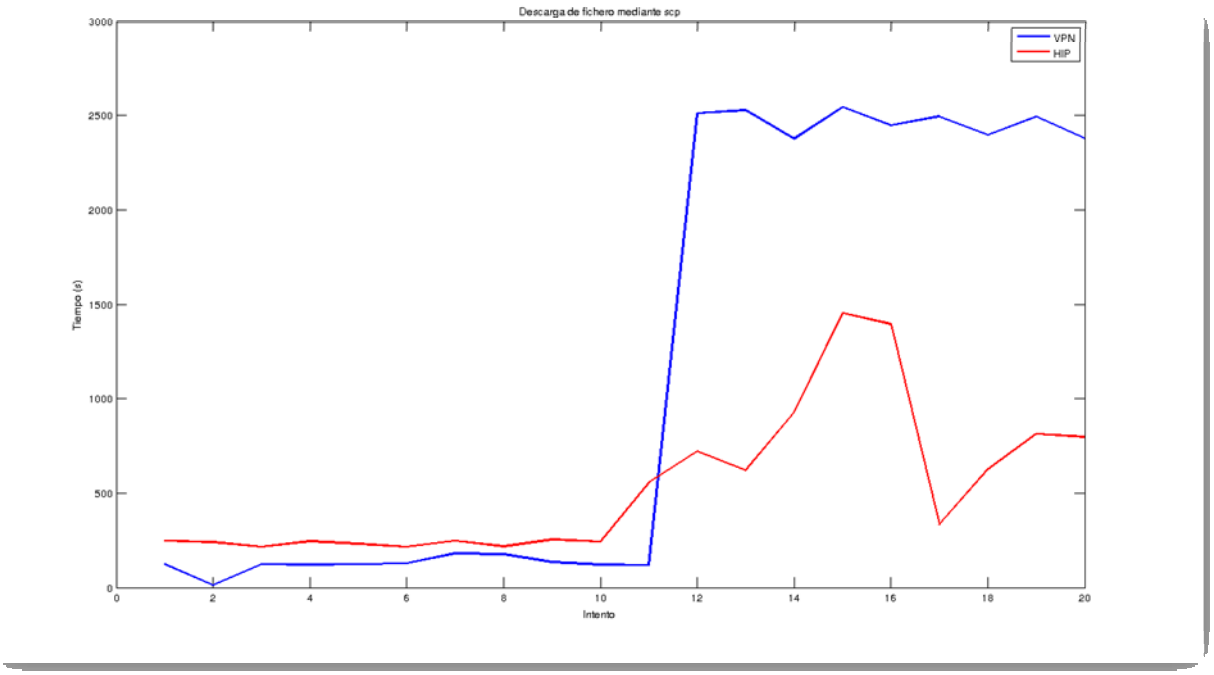


Ilustración 42 : Comparativa con SCP entre OpenVPN & HIP

En cuanto a la transferencia de un fichero mediante el comando scp, la novedad radica en que HIP sufre cierto aumento del tiempo de respuesta, pero aún así se mantiene por debajo de los tiempos de respuesta medidos en las pruebas de OpenVPN.

Descarga de ficheros mediante FTP

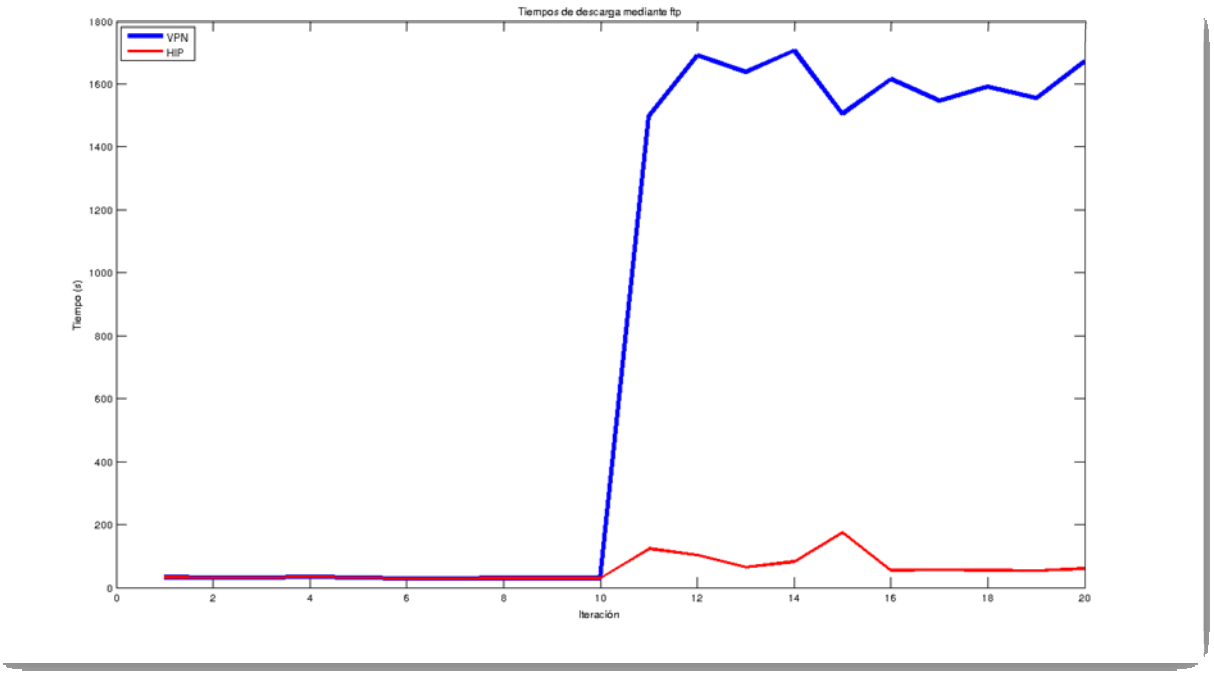


Ilustración 43 : Comparativa con FTP entre OpenVPN & HIP

En el caso de la transferencia mediante FTP, no hay nada nuevo respecto a las anteriores mediciones en otras pruebas. Nuevamente VPN muestra tiempos muy superiores a los de HIP.

Escaneo de puertos mediante hping

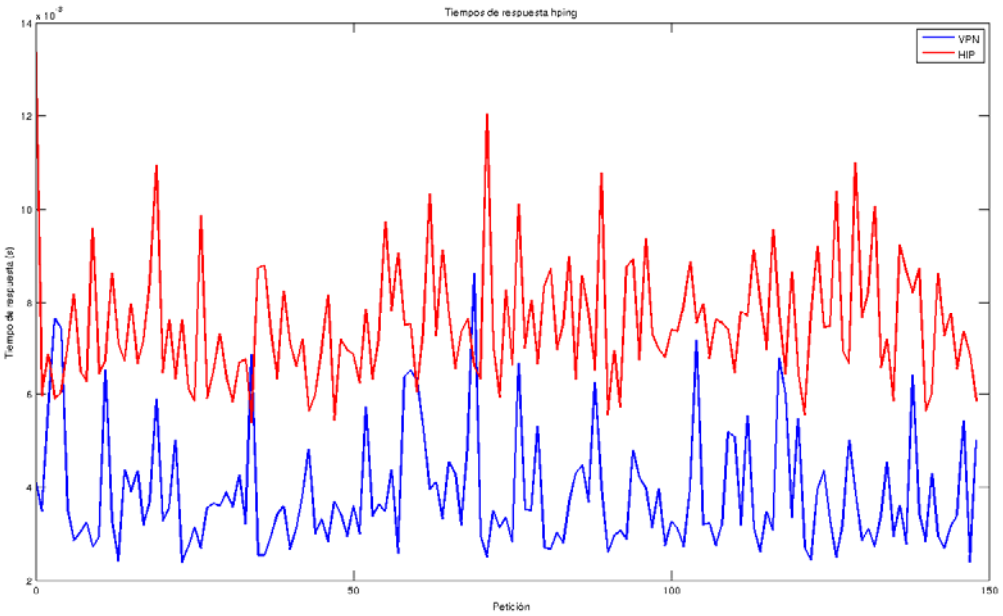


Ilustración 44 : Comparativa con HPING entre OpenVPN & HIP

En este último caso, sí que se ha dado un resultado no encontrado en todas las pruebas anteriores. En las pruebas de escaneo de puertos (entorno virtual) OpenVPN se ha mostrado más rápido que HIP. Se puede ver que hay bastante variación dentro de cada una de las tecnologías, pero si se comparan las trazas entre sí, puede verse claramente que HIP se mantiene en todo momento con tiempos más altos que OpenVPN

PROBLEMAS ENCONTRADOS

Problemas con KVM

En un principio, el profesorado aconsejó al grupo de trabajo para que se utilizara KVM como solución para la virtualización, ya que todas las pruebas del proyecto serían realizadas en máquinas virtuales.

El grupo de trabajo comenzó por la vía de utilizar KVM pero no se consiguió que el software funcionara de forma fluida. Concretamente, el problema fue que las máquinas virtuales se ejecutaban de forma demasiado lenta y esto impedía una ejecución mínimamente aceptable para realizar ningún trabajo.

Después de numerosos intentos, se llegó a la conclusión de que era mejor cambiar de software que investigar el problema que estaba ocasionando tal ineficiencia a la hora de ejecutar las máquinas virtuales. Teniendo en cuenta que algunos de los componentes del grupo de trabajo tenían experiencia en el uso de máquinas virtuales con Virtualbox, se optó por utilizar dicho software.

Sistema Operativo de 64 bits

Una de las máquinas con las que se ha desarrollado el proyecto, el Lenovo g550, tiene un sistema operativo Windows 7 home Premium 64 bits. Al proceder a la instalación de Virtual Box en este equipo, no hubo ningún problema. Pero después de familiarizarse con el programa de virtualización, y de empezar a hacer pruebas con las máquinas, se observó que había un problema, y es que las redes no funcionaban del todo bien. Cuando solo se trabajaba en una red no había problema, pero al crear dos redes y un una máquina virtual con dos interfaces de red, e intentar que propagara los paquetes, se creaban problemas, y al apagar la máquina, y volver a su configuración, esta no estaba con la configuración que se había dejado al principio. El problema fue que pasó un tiempo desde que se empezó a trabajar y a hacer pruebas hasta que se detecto el fallo, y fueron días de trabajo perdido. Después, buscando en internet, en un foro, se encontró el problema, que Virtual Box no está preparado para sistemas operativos de 64 bits. Se puede instalar y hacer cosas, pero no se garantiza la funcionalidad 100%, sino que hay cosas que pueden fallar, como así sucedió.

Imposible hacer funcionar Mobile IP

En un principio, el proyecto consideraba la evaluación de tres tecnologías de movilidad IP: VPN, HIP y MIP.

En el caso de la última, no fue posible hacer funcionar la infraestructura creada utilizando MIP, ya que fue imposible simular la movilidad de los nodos móviles y que el sistema respondiera como se esperaba.

Dado además que Mobile IP para IPv4 está prácticamente abandonado y que este proyecto no se centra en IPv6, no existe apenas documentación que ayude a poner en marcha un sistema de estas características en IPv4.

Viendo todo lo anterior y después de pasar un tiempo considerable intentando buscar los problemas que causaban la mencionada imposibilidad de hacer funcionar el sistema, el grupo de trabajo y el profesor llegaron a la conclusión de que lo mejor era centrarse en las otras dos tecnologías y dejar Mobile IP a un lado.

No reconoce el cambio de red hasta pasados unos paquetes

Otro de los problemas se encontró al cambiar de red mientras se ejecutaba un ping, y ejecutar el cambio de openvpn, fue que, aunque se completaba la conexión a la otra red y se inicializaba correctamente openvpn, la conexión del ping entre los dos pc's tardaba muchos paquetes más en volver a conectarse. En cambio, si durante esa espera, se lanzaba un ping desde el otro pc al que había lanzado el primero, en el momento, había ping, y además, el primer ping, el que estaba a la espera, se ponía en marcha otra vez. Después de investigar, se descubrió que el problema estaba en que la MAC de la tarjeta de red, que al cambiar de red el pc y utilizar el tap, cambiaba la mac de la tarjeta de red, y el otro pc no era capaz de darse cuenta que ese ordenador ya estaba otra vez conectado. Se hallaron dos soluciones posibles: la primera era, una vez cambiada la red, lanzar un brodcast para que el pc se diera cuenta que ya se había vuelto a conectar, pero no era posible, por lo que se optó por la segunda opción, que era una vez que se había cambiado de red, y como se utilizaba otra interfaz distinta, cambiar la mac de esa tarjeta por la mac de la primera. Se creó un script para que el cambio de red y el cambio de mac fuesen automáticos, y se consiguió el objetivo. En cuanto se lograba tener conexión a través de openvpn, se restablecía la conexión del ping, sin necesidad de hacer nada más.

Fallo al iniciar el bridge

Con openvpn ya instalado, y funcionando, al apagar las máquinas y volver a encenderlas, se tuvo un problema. El primer paso para poder hacer funcionar openvpn con la configuración de las máquinas era levantar el bridge en el servidor openvpn. Pero al iniciar el bridge en el servidor aparecía un error, diciendo que el tap estaba utilizado. Inicialmente, la forma de solucionarlo fue cambiar la configuración, y dar otro nombre al bridge y al tap, pero no era una solución muy profesional. Después de investigar, se encontró el error, y fue que al iniciar el pc, openvpn se auto iniciaba, y cogía los recursos tap0 y br0. Matando el proceso (pkill openvpn), los recursos se liberaban y ya se podía iniciar openvpn sin problemas. Finalmente se cambió la configuración, para que openvpn no se iniciara al arrancar el pc, y se solucionó el problema sin necesidad de matar el proceso.

CONCLUSIONES

El siguiente apartado tratará de dar las conclusiones a las que se ha llegado en el proyecto. Obviamente, las conclusiones serán tanto técnicas como personales. En el caso de las técnicas, se intentará dar respuesta al problema planteado como objetivo del proyecto, es decir, dar una visión razonada de qué protocolo de los analizados se comporta mejor en la mayoría de los casos, y por tanto, cuál de ellos está más preparado para hacer frente a escenarios de movilidad como los expuesto a lo largo del presente documento.

Por otra parte, en las conclusiones personales se dará la opinión que el grupo tiene de todo lo relacionado con el proyecto, desde conocimientos adquiridos hasta la experiencia que ha supuesto el desarrollo del proyecto.

Conclusiones técnicas

Al querer comparar estas dos tecnologías y querer sacar una conclusión definitiva de qué tecnología es superior, nos encontramos con el obstáculo de que las dos son muy similares y ninguna destaca sobre la otra.

Veamos cómo se comportan las dos tecnologías en un entorno virtual, es decir, en un entorno ideal, puesto que no tenemos retardos por culpa del medio:

Se han hecho 6 pruebas distintas usando las dos tecnologías y se ha medido el tiempo de reconfiguración de cada tecnología, es decir, el tiempo que tarda en volver a funcionar la prueba que se estaba haciendo, y la proporción de más que han necesitado las dos tecnologías para realizar la transferencia cuando los servidores se trasladaban a la otra red.

Las conclusiones cuantitativas finales están muy claras: HIP se comporta extremadamente bien una vez que el servidor se cambia de red. El tiempo de transferencia de las distintas pruebas apenas varía de cuando el servidor se encuentra en la misma red que el cliente. En cambio VPN, en casi todas las tecnologías, el retardo que ha sufrido ha sido muy grande y en algunas pruebas como SCP o descargas HTTP, ha tardado 54 y 67 veces más que si el servidor no se hubiera cambiado de red.

Pero por otro lado, si nos fijamos el tiempo que las dos tecnologías tardan para que las distintas pruebas vuelvan a estar operativas, VPN tiene mejores tiempos. Sus tiempos varían entre 5 y 13 segundos de reconfiguración, HIP en cambio ha necesitado de media 12-20 segundos.

Estos datos son muy significativos pero veamos cómo se comportan en un entorno real:

El tiempo de transferencia sigue siendo menor para HIP y es muy parecida al entorno virtual, pero resulta que VPN mejora sus tiempos en un entorno real. Las transferencias de ping y peticiones web tardan el doble y las descargas http y las pruebas scp, que antes tardaban 54 y 67 veces más, ahora la proporción es de 21 y 18.

Por otro lado, el tiempo de reconfiguración que antes VPN obtenía los mejores tiempos, ahora HIP es más rápido reconfigurándose. Ha aumentado sus anteriores tiempos de 12-20 segundos a 20-34 pero ese aumento de tiempo es mucho menor que VPN, que

sube desde los 5-13 segundos anteriores a medias desde 19 segundos hasta el minuto y 49 de la prueba de scp.

Los datos que antes hemos conseguido anteriormente se han conseguido a raíz de que cada tecnología ofrece unas ciertas ventajas y por ese motivo añade más datos a los datagramas o lo paquetes tienen que pasar por una serie de procesos que hacen que la comunicación dure más.

VPN añade un encapsulamiento en las capas 2 y 3 para que las transferencias funcionen también con protocolos que no son IP y para añadirle seguridad mediante IPsec. El método de IPsec que usamos es el modo túnel y para poder enviar los paquetes por el túnel han de ser encapsulados y desempaquetados en el otro extremo.

El modo de funcionar de HIP hace que la reconfiguración sea más costosa pero después añade menos encapsulamiento con lo que la transferencia de archivos es más ligera. Hip también usa IPsec y el encapsulamiento ESP, pero el inicio de la comunicación con los dos end-points pasa por una comunicación de ida y vuelta, mediante los anteriormente mencionados i1, r1, i2 y r2. En este intercambio de datos se usa el algoritmo de Diffie Helman para evitar posibles ataques, en el que el host remoto debe resolver el problema de codificación y que cada vez que falla, aumenta la dificultad del problema a solucionar. La autenticación se consigue con el par de claves que los dos extremos consiguen y la criptografía de ello requiere tiempo de cómputo.

Vemos que si quisiéramos usar estas tecnologías para no perder conectividad en ningún momento probablemente usaríamos HIP: es más rápido una vez cambiado de red y no tiene mucho que envidiar en la velocidad de reconfiguración si comparamos con VPN.

¿Pero cuáles son las ventajas y los contras de estas dos tecnologías?
VPN provee seguridad, estabilidad y comprobados mecanismos de cifrado sin sufrir la complejidad de otras soluciones VPN como las de IPsec. Ofrece también la posibilidad de de implementar dos modos básicos en capa 2 o capa 3 con lo que somos capaces de enviar información en otros protocolos no-IP y también ofrecer soporte proxy.

OpenVPN soporta también una amplia gama de modos de configuraciones, incluyendo acceso remoto, sitio a sitio VPN, Wi-Fi, balanceo de carga y conmutación por error.

HIP en cambio propone una nueva capa en la pila de protocolos, entre los niveles de red y de transporte. Este nuevo nivel proporciona autenticación, con independencia de la dirección IP del host. Además, con el protocolo HIP se establece una asociación segura entre los dos hosts que participan en la comunicación. Esta asociación, que se crea como paso previo al intercambio de datos, proporciona autenticación, integridad, confidencialidad de los datos enviados y es resistente a ataques de Denegación de Servicio (DoS) y a ataques de reenvío de paquetes.

HIP se encarga de que los usuarios no pierdan la identidad que poseen aunque cambien de ubicación en la red o cambien de dirección IP. Estos cambios son totalmente transparentes para el usuario si utiliza HIP.

HIP también añade características multi-homing de movilidad a los dispositivos habilitados por HIP, es decir, que se tiene más de una conexión simultánea a Internet. En HIP un nodo es capaz de recibir paquetes creados por HIP de cualquier dirección IP. Un nodo

puede cambiar su ubicación topológica y continuamente enviar y recibir paquetes desde y hacia sus compañeros

Estas tecnologías nos permiten seguir conectados a una red en la que físicamente no lo estamos y es una forma de abordar el problema de la movilidad en internet. A la hora de comparar estas dos tecnologías y dar un veredicto, saltan las dudas puesto que cada tecnología tiene sus pros y sus contras. Open VPN tiene unas velocidades de reconexión muy buenas comparadas con hip: Open vpn ronda los 9-16 segundos en máquinas virtuales y 20-30 en la infraestructura real cuando hip dobla estas velocidades. Si es cierto que la velocidad de Open VPN en transferencia de archivos grandes deja mucho que desear: por ejemplo un archivo de 200 megas, que equivaldría a un video en internet, se descargaría a 100 KB/s, una velocidad muy lenta hoy en día si lo comparamos con la banda ancha. Hip en cambio nos ofrece velocidades de entre 300 y 700 KB/s, algo más aceptables. La perdida de paquetes también es inferior si usamos el protocolo HIP. Se pierden entre 1 y 3 paquetes en la mayoría de pruebas mientras que Open vpn puede llegar a perder 7 paquetes en un entorno real. Como hoy en día todo se rige por la velocidad y el tiempo vale oro en internet, a las dos tecnologías les hace falta un empujón, por eso mismo dejamos que ustedes prueben que tecnología les gustan más.

Aún así, en principio nos decantamos más por HIP, ya que como se ha dicho anteriormente consigue mayores velocidades a pesar del hándicap a la hora de realizar los cambios de red. Nos parece que no tiene sentido utilizar VPN y esgrimir como ventaja el hecho de que los cambios de red se den más rápidamente, si después no se va a poder una conexión aceptable que posibilite casi cualquier tarea en la red de hoy en día.

Como conclusión, por tanto, nos gustaría destacar que ambas tienen sus ventajas e inconvenientes, pero que si hay que decantarse por una nos quedamos con HIP.

Conclusiones personales

En el presente apartado se intentará plasmar la opinión que los componentes del grupo de trabajo que ha desarrollado el proyecto. Dicha opinión será una opinión personal de todo lo acontecido a lo largo del año que ha durado el proyecto, tanto sobre los conocimientos adquiridos como de la experiencia vivida en cuanto al trabajo en grupo realizado y el hecho de tener que hacer frente a un proyecto de un tamaño más grande que los que se acostumbran a tener a lo largo de la carrera.

En primer lugar, hay que decir que la temática del proyecto ha sido en cierto modo diferente a lo que se ha hecho en años anteriores, lo cual ha servido de herramienta muy valiosa para el aprendizaje en un tema como son las redes. Los componentes del grupo pensamos que como ingenieros que queremos ser, el campo de las redes es un tema que debemos dominar y es precisamente esto lo que hemos conseguido con el proyecto. Si bien ya habíamos cursado la asignatura de redes, este proyecto nos ha puesto ante un reto que escapaba de los conceptos teóricos cursados en dicha asignatura, para adentrarnos en un entorno más práctico. Podemos decir que hemos aprendido a configurar y poner en marcha routers, establecer infraestructuras VPN y HIP e incluso hemos obtenido conceptos teóricos de Mobile IP, a pesar de no haber podido desarrollar este protocolo como nos hubiera gustado.

En lo personal, nos hemos visto en la necesidad de trabajar en grupo, organizando las tareas a realizar para cada integrante del equipo y trabajando para que el trabajo en paralelo se sincronizara de la mejor manera posible. Queremos decir que esta necesidad nos ha dado más experiencia aún en trabajar en grupo con todas las ventajas e inconvenientes que ello conlleva, pero también queremos hacer un poco de autocrítica, ya que pensamos que en algunas fases del proyecto la organización del grupo podría haber sido mejor. Pese a esto, estamos satisfechos del trabajo realizado y en general creemos que hemos sido capaces de hacer frente a las dificultades que se nos han planteado.

Además del trabajo en grupo, la propia duración del proyecto ha sido un escollo más en este tiempo, ya que al no ser un período parecido al de prácticas que hayamos podido realizar en anteriores cursos, nos ha sido más complicado dimensionar en el tiempo todas las tareas que teníamos para hacer.

Para terminar con estas conclusiones personales, por tanto, nos gustaría hacer hincapié en que nos ha parecido un proyecto que se ha alejado de un proyecto más “típico”, como podría ser el desarrollo de un software. Sin intentar menospreciar a este tipo de proyectos, queremos resaltar que la temática en la que se ha desarrollado el nuestro nos ha permitido aprender sobre un tema que nos parece muy importante, y que a lo largo de la carrera no hemos tratado tan a fondo como otras materias.

POSIBLES MEJORAS Y AMPLIACIONES

El grupo de trabajo piensa que el objetivo del proyecto se ha alcanzado, pero aún así tiene en mente algunos puntos que le hubiera gustado desarrollar o que le parecen interesantes para tratar de ampliar el proyecto. Este apartado hablará de esos puntos que el grupo considera interesantes para ser tratados en un futuro, en el caso de que se quisiera proseguir con el desarrollo del proyecto.

Trabajar a través de la red real

Como el lector ya habrá podido observar, las pruebas realizadas se han limitado a pruebas en redes de área local y redes comprendidas dentro de una sola máquina, mediante el uso de máquinas virtuales. A pesar de que estas pruebas realizadas han servido para responder a la problemática del proyecto, realmente los resultados obtenidos no pueden aplicarse a usos en entornos reales.

Para poder dar una visión clara de cómo se comportarían los protocolos estudiados en un entorno real, se considera indispensable hacer las pruebas a través de Internet. De esta forma se podría ver el comportamiento de los sistemas en un entorno no controlado, por lo que daría una visión más exacta de la realidad.

Esta ampliación es, posiblemente, la más difícil de realizar, ya que entrarían en juego sistemas reales y harían falta elementos como IPs públicas y capacidad de mover los equipos de una red a otra. El realizar todo esto, en principio, no es trivial, pero no imposible, por lo que en caso de que pudiera llevarse a cabo sería una ampliación muy interesante.

Probar otras implementaciones de los estándares

Es bien sabido que en el mundo de la informática los estándares son una cosa, y sus implementaciones otra, pudiendo existir diferencias más que notables entre implementaciones del mismo estándar.

Es por ello por lo que se cree necesario probar otras implementaciones diferentes a las evaluadas en este proyecto, porque los resultados pueden ser muy diferentes.

Por poner un ejemplo, en vez de utilizar OpenVPN, podría hacerse uso de otras soluciones más enfocadas a conseguir altas tasas de transferencia. Existen multitud de soluciones de pago que consiguen un mejor rendimiento que OpenVPN, por lo que podría hacer variar las conclusiones a las que se ha llegado en este proyecto.

Lógicamente, lo mismo sucede con HIP.

Probar con el estándar IPv6

El proyecto se ha desarrollado bajo el estándar Ipv4. Este estándar está en proceso de reemplazo en favor de la versión 6. A pesar de que el cambio tardará unos años y el estándar Ipv4 seguirá usándose todavía durante este tiempo, IPv6 es una realidad cada día más presente en la red. Esto hace prácticamente obligado el testear los protocolos en IPv6, ya que será la única forma de ver el comportamiento de los protocolos evaluados en la red del futuro.

Por otra parte, sería posible añadir a la evaluación el protocolo Mobile IP, ya que este estándar está más vivo en IPv6 que en IPv4.

Probar sobre otros sistemas operativos y con diferente hardware

Para la realización de las pruebas, el grupo de trabajo ha utilizado los ordenadores personales de los que disponía, y por tanto no se pudo hacer una selección de hardware para trabajar. Estas limitaciones no han supuesto mayor problema puesto que hoy en día los ordenadores de la amplia mayoría de los usuarios tienen unas prestaciones bastante similares. Sin embargo, parece interesante probar los protocolos utilizando diferente hardware, como por ejemplo tarjetas de red tanto de gama alta como de gama baja. Con esta ampliación se podría concluir si los protocolos testados son muy dependientes de la calidad de los dispositivos utilizados o se comportan de forma similar en unos y otros.

También, a la hora de hacer las pruebas, sería conveniente poder utilizar las mismas máquinas para las mismas pruebas, ya que así se estaría asegurando que el hardware no sería un elemento diferenciador de cara a los resultados.

En cuanto al software, por otro lado, si que se pudo hacer una selección de los sistemas que se utilizarían, pero más que nada las elecciones se guiaron por lo conocimientos del profesorado en este sentido. Probar los protocolos en otros sistemas operativos es un punto que no estaría de más, aunque a priori los resultados obtenidos no deberían ser muy diferentes a los presentados en la presente memoria.

ÍNDICE DE ILUSTRACIONES

ÍNDICE DE TABLAS

GLOSARIO

- **Base exchange –** Protocolo criptográfico
- **Clave privada -** La clave privada o secreta de un par de claves criptográficas asimétricas. Se supone que se conoce sólo la parte identificada con la clave pública correspondiente. Utilizado por la parte identificada para autenticar su identidad a otras partes.
- **Clave pública -** La clave pública de un par de claves criptográficas asimétricas. Se utiliza como un identificador público para la autenticación de la identidad criptográfica.
- **Diffie-Hellman –** protocolo que permite el intercambio secreto de claves entre dos partes que no han tenido contacto previo, utilizando un canal inseguro, y de manera anónima
- **DNS – Domain Name Service.** Un sistema que resuelve nombres inteligibles para los humanos en identificadores binarios asociados con los equipos conectados a la red.
- **DHCP** – **Dynamic Host Configuration Protocol** - protocolo de red que permite a los clientes de una red IP obtener sus parámetros de configuración automáticamente.
- **Encapsulating Security Payload** (**ESP**) proporciona confidencialidad y la opción -altamente recomendable- de autenticación y protección de integridad.
- **End-point** - Una entidad de comunicación
- **FTP – File transport protocol -** es un protocolo de red para la transferencia de archivos entre sistemas conectados a una red TCP (Transmission Control Protocol), basado en la arquitectura cliente-servidor
- **HI – Host identity**, Un concepto abstracto asignado a una plataforma de computación
- **HIP** – **Host Identity Protocol**, Un protocolo usado para transportar y autenticar los HI y otra información.
- **HIT – Host Identity Tag**, Un dato de 128 bits creado mediante código criptográfico apliciado al HI.
- **Host** – computadora conectada a una red
- **HTTP – Hypertext Transport Protocol -** protocolo usado en cada transacción de la World Wide Web
- **ICMP – Internet Control Message Protocol -** sub protocolo de control y notificación de errores del Protocolo de Internet (IP)
- **IP – Internet protocol,** referring to either IPv4 or IPv6 - es un protocolo no orientado a conexión usado tanto por el origen como por el destino para la comunicación de datos a través de una red de paquetes conmutados no fiable de mejor entrega posible sin garantías
- **IPsec – Internet Protocol Security -** es un conjunto de protocolos cuya función es asegurar las comunicaciones sobre el Protocolo de Internet (IP) autenticando y/o cifrando cada paquete IP en un flujo de datos
- **IPv4 – Internet Protocol version 4**
- **IPv6 – Internet Protocol version 6**
- **LAN – Local Area Network -** es la interconexión de varias computadoras y periféricos.
- **LSI – Local Scope Identifiers,** Un dato de 32 bits que indica una identidad de host

- **MIP – Mobile Internet Protocol,** referring to either MIPv4 or MIPv6 - es un protocolo estándar diseñado para permitir a los usuarios de dispositivos móviles moverse de una red a otra manteniendo permanentemente su dirección IP.
- **Multi-homing** - is a technique used to increase the reliability of the Internet connection for an IP network.
- **NAT – Network Address Translation -** es un mecanismo utilizado por enrutadores IP para intercambiar paquetes entre dos redes que se asignan mutuamente direcciones incompatibles
- **Par de claves pública** - Un par de claves criptográficas asimétricas que consta de claves públicas y privadas.
- **PEERS** – par, igual, se refiere a máquina o host
- **SA** – asociación segura
- **SCP** – Copia segura
- **SFTP** – Secure File Transfer Protocol
- **SSH** – Secure Shell - es el nombre de un protocolo y del programa que lo implementa, y sirve para acceder a máquinas remotas a través de una red.
- **TCP** – Transmission Control Protocol
- **UDP** – User Datagram Protocol - es un protocolo del nivel de transporte basado en el intercambio de datagramas.
- **VPN** – Virtual Private Network - es una tecnología de red que permite una extensión de la red local sobre una red pública o no controlada, como por ejemplo Internet.
- **WLAN** – Wireless Local Area Network - es un sistema de comunicación de datos inalámbrico flexible, muy utilizado como alternativa a las redes de área local cableadas o como extensión de estas.

BIBLIOGRAFÍA

Libros

Internet

Wikipedia

- Mobile IP. 17-01-2011
  - Castellano: http://es.wikipedia.org/wiki/IP_M%C3%B3vil
  - Inglés: http://en.wikipedia.org/wiki/Mobile_IP
- Internet
  - http://es.wikipedia.org/wiki/Internet
  - http://es.wikipedia.org/wiki/Red_de_comunicaciones
- Modelo OSI
  - http://es.wikipedia.org/wiki/Modelo_OSI

Dynamics Mobile IP

- Página del Proyecto. 17-01-2011. http://dynamics.sourceforge.net/
- SourceForge. 17-01-2011. http://sourceforge.net/projects/dynamics/

VirtualBox

- Web Oficial. 17-01-2011. http://www.virtualbox.org/
- Documentación. 17-01-2011. http://www.virtualbox.org/manual/

Ubuntu

- Documentación. 17-01-2011.
  - KVM. https://help.ubuntu.com/community/KVM

RFCs

- Están en el documento de Mobile IP

Openvpn

- Web Oficial. -----. http://openvpn.net/
- Documentación. ----
  - How to install - ----- - http://openvpn.net/index.php/open-source/documentation/howto.html
  -

HIP

- Web Oficial. -----. http://www.openhip.org/
- Documentación. ----
  - http://www.ietf.org/rfc/rfc4423.txt?number=4423

- http://hipl.hiit.fi/index.php?index=how
- http://tools.ietf.org/html/rfc5201
- http://voiplab.niu.edu.tw/ppt/ipv6/R9843010.pdf
- http://koti.welho.com/pnikande/publications/hip_survey.pdf

RESUMEN

Introducción, problemática y objetivo

Desde sus inicios, las redes de computadores han ido avanzando en sus implementaciones con el fin de proporcionar unos niveles de seguridad adecuados y una velocidad de transmisión que permita que la comunicación sea viable.

La red de computadores más conocida y utilizada hoy en día es sin duda Internet, una red que a nivel global es capaz de poner en contacto computadores sin importar en qué lugar se encuentren.

Obviamente, con el paso de los años, las necesidades que las redes de comunicación debían cubrir han ido cambiando, pasando de ser un mero enlace de documentos de hipertexto a convertirse en la vía de transmisión de todo tipo de contenidos. Uno de esos cambios más importantes sobre los que se está trabajando es en la movilidad de los equipos en internet.

Uno de los protocolos más usados en internet es el protocolo cliente – servidor. En él, existe un servidor que ofrece datos y servicios a otras máquinas, que ejercen el papel de cliente. Existen numerosas posibilidades para tratar el problema de movilidad en los clientes. Sin embargo, este proyecto se centra en un punto de vista diferente. Este punto de vista considera que el elemento que está en movilidad es el servidor en vez del cliente. El tener el servidor en movilidad es un escenario que no es tan usual como el anterior, y por eso ha sido el escenario objeto de investigación del proyecto.

El uso de servidores en la actualidad es extendido, por todos los usuarios, ya sea a nivel de empresa para trabajar sobre datos y ficheros con datos comunes, como para consultar información de periódicos o revistas colocadas en servidores, hasta descargar canciones o películas, también almacenadas en servidores. Todo este tipo de interacción entre máquinas e intercambio de información tienen un patrón común: la existencia de un servidor, en el que se almacena la información, y de unos clientes, que hacen uso de ella. Puesto que mucha gente depende de un servidor, es un tema delicado un cambio de red del servidor. Este cambio se puede deber a diversos motivos: sobrecarga de la red en la que se encontraba, balanceo de carga por parte del firewall o router para optimizar la velocidad de la conexión, avería en la red en la que estaba y traslado a otra red de la empresa pero con distinta dirección…. En todos los casos, la dirección IP del servidor tendrá que cambiar. ¿Qué problemas acarrea esta situación?

La dirección IP se encarga de otorgar a la máquina identificación y ubicación. Si se cambia la dirección IP de un servidor para poder colocarlo en otra red, se cambiará su ubicación, que era el objetivo; pero también se estará cambiando la identificación del servidor, por lo que, todos los clientes que tuvieran conexiones realizadas con el servidor no sabrán donde se encuentra este, por lo que la conexión que se interrumpió no podrá ser reanudada. Para que estas conexiones no se pierdan, y solo se detengan momentáneamente durante el cambio de red del servidor, existen diversas técnicas: VPN a nivel software, VPN a nivel hardware, MIP, HIP,….; en todas ellas, el objetivo es, de una forma o de otra, que esa conexión no se pierda. Consiguen que, aunque una máquina se traslade a otra red, la conexión que había con la IP antigua no se pierda, y de una forma u otra, el cliente sea capaz

de conocer la nueva ubicación del servidor, y reanudar la transferencia. Sobre esta problemática tratará el proyecto.

El objetivo del proyecto es la evaluación y comparación de tecnologías de movilidad IP sobre servidores. Dada la problemática con el cambio de IP de los servidores mencionada anteriormente, y el extendido e imprescindible uso de los servidores por parte de todos los usuarios de la red, es necesario poder migrar un servidor de red sin necesidad de perder las conexiones que se tuviera en ese momento, y hacer que el tiempo que tarde en restablecerse la configuración del mismo, así como en levantarse las conexiones sea mínimo.

Por estos motivos surge este proyecto. Dado que hay tecnologías encargadas de esta movilidad, se quiso hacer un estudio detallado de estas, con el objetivo de evaluar y comparar dichas tecnologías.

Existen diversas tecnologías para la movilidad IP aplicables a servidores. En este trabajo se optó por estudiar las dos técnicas más desarrolladas y avanzadas que existen en este sector, como son HIP y VPN, pues son de las más extendidas en el mercado, las más comunes, y son diversas entre ellas, y se creyó que era una buena elección para poder tener una buena comparativa.

Para evaluar VPN, se optó por OPENVPN, software gratuito de SSL, que consiste en montar una red virtual encima de la red pública. Con ello, y a través de diversos mecanismos que se explican a continuación, se puede tener un equipo (en este caso un servidor) en una red física, pero que “virtualmente” esté en otra red, teniendo una IP de la misma, y teniendo conexión a través del servidor VPN.

Con la tecnología HIP decidimos utilizar OPENHIP, también software gratuito de SSL. Su estándar más moderno y avanzado es la versión 7.0, y fue sobre la que se desarrolló el proyecto. HIP consiste en introducir una nueva capa al protocolo TCP/IP, la capa HIP. Con esta nueva capa, se consigue desligar la identidad y localidad que lleva consigo la dirección IP; así, ahora obtenemos dos direcciones, una con la que se indica la identidad de cada máquina, la cual no cambia, y otra que indica la localidad y ubicación de esta máquina (en este caso, el servidor, que es sobre lo que se desarrolla el proyecto). Con ello, cuando se produce un cambio de red del servidor, la última dirección cambiará, y mediante una serie de actualizaciones, los clientes sabrán el lugar donde se encuentra, por lo que un cambio de ubicación no será algo definitivo para la comunicación, como ocurría con un cambio de la dirección IP.

VPN

Para realizar las pruebas de vpn, instalamos en las maquinas virtuales OpenVPN

OpenVPN es una aplicación de software de código libre y abierto que implementa red privada virtual (VPN) para crear soluciones de seguridad punto a punto o conexiones de sitio a sitio (http://www.openvpn.net/), una aplicación de software gratuito, de código libre y abierto, que implementa vpn. Se implementa en las capas 2 y 3 del modelo OSI, usa protocolos SSL/TLS, y se basa en la certificación del cliente en el servidor mediante una serie de certificados. No es un proxy de aplicación web, ni trabaja a través de un navegador web, sino que crea soluciones de seguridad punto a punto, o conexiones a sitio. Para ello, crea una interfaz en el cliente, y le asigna una IP de la red privada del servidor, con lo que ahora el cliente puede trabajar con los recursos locales, tales como impresoras, servidores…, sin necesidad de estar en la red físicamente.

Para poder llevar a cabo la conexión, es necesaria la autentificación y autorización del usuario y el equipo sobre el servidor de vpn. Los requerimientos para esta conexión son:

- Identificación del usuario
- Codificación de los datos (mediante algoritmos cifrados)
- Actualización de las claves por parte del servidor VPN para los usuarios

Existe una red central (servidor) y unas oficinas remotas (clientes). Se pueden dar distintos permisos a distintos usuarios, los cuales se tienen que autentificar contra el servidor.

Cada oficina tiene su propia LAN, y un túnel distinto. El servidor VPN hace de pasarela; ahora queda como una red local, ya que él captura el tráfico que va hacia sus clientes. Los clientes están conectados por una red virtual proporcionada por el servidor.

Para la configuración de openvpn en el proyecto, se decidió por utilizar un bridge en el servidor, y un tap para poder conectarse a él. Un *bridge* (o puente) es un dispositivo de interconexión de redes de ordenadores, que actúa en la capa de datos. Sirve para conectar dos segmentos de red como si fueran una única. Un *tap* es una interfaz que se crea para hacerse ver en una red virtual, para que la máquina con el tap se pueda ver en la red en la que se autentifica como si estuviera en ella.

Una vez explicado lo que es un bridge y un tap, se explica el funcionamiento que tienen utilizando la tecnología openvpn. En el servidor se crea el bridge y todos los clientes que quieran se conectan a él utilizando cada uno una interfaz tap, que sería el equivalente a que se conectasen a un switch virtual. Para cada uno de estos clientes, el servidor crea un túnel a través del cual irán encaminados los paquetes a y desde los clientes. De esta forma, todos los paquetes que se envíen deben encapsularse para que pasen a través del túnel, lo que genera una mayor carga de información a enviar.

Gracias a esta estructura, cuando una máquina se encuentra en una red, y quiere conectarse por vpn a otra red, sólo debe iniciar openvpn, y el servidor le otorgará, a través del tap, una IP de la red a la que quiere acceder, teniendo la máquina dos direcciones IP, la suya de su red, y la que le ha otorgado el servidor vpn en el tap. Así, ahora esta máquina está identificado con las dos direcciones, y a través de ambas se puede establecer conexión con ésta máquina.

HIP

HIP es un protocolo para Internet que permite establecer conexiones seguras entre hosts, y mantener estas conexiones aunque la localización (dirección IP) de los hosts cambie.

Esto se consigue con una capa en la pila de protocolos, en particular entre las capas de red y transporte. Durante el establecimiento de la conexión, los hosts intercambian sus claves públicas y acuerdan una clave secreta de sesión. HIP asume que inicialmente se conoce (o se puede consultar en un DNS) la IP y la identidad del host con el que se quiere establecer la comunicación.

HIP se encarga de que los usuarios no pierdan la identidad que poseen aunque cambien de ubicación en la red o cambien de dirección IP. Estos cambios son totalmente transparentes para el usuario si utiliza HIP.

Hip introduce un nuevo intercambio criptográfico, el host identity Base Exchange. Este nuevo intercambio permite a los peers establecer SA-s utilizados por IPsec para la creación de una conexión punto a punto seguro. El tráfico de datos de esta conexión está asegurado mediante los paquetes ESP. En la actualidad, ESP es la mejor manera de proteger la carga en un datagrama. HIP también añade características multi-homing y de movilidad a los dispositivoshabilitados por HIP habilitado.

HIP es, básicamente, un intercambio de claves de ida y vuelta de Diffie-Hellman, un "Base exchange", y algunos mensajes adicionales.La "Base exchange" se hace para confirmar que dos peers tienen sus propias claves privadas correspondientes a sus propias HI-s, que son las claves públicas. Cuando un host se ha autenticado el "Base exchange" establece dos SA-s para que la seguridad punto a punto sea encapsulada por ESP.

El flujo del "Base exchange" se puede describir en los siguientes pasos:

Iniciador -> Directorio: buscar respondedor
Iniciador <- Directorio: El respondedor responde con dirección y HI / HIT
I1: Iniciador -> Respondedor (Hola, Este es mi I1, vamos a hablar con HIP)
R1: Respondedor -> Iniciador (Ok, aquí está mi R1, manejar esta cookie HIP)
I2: Iniciador -> Responder (Computando, aquí está mi contador I2)
R2: Responder -> Iniciador (OK. Vamos a terminar HIP con mi R2)
Iniciador -> Responder (Datos protegidos por ESP)
Responder -> Iniciador (Datos protegidos por ESP)

Explicación de la prueba

Una vez claro el problema (evaluación y comparación de tecnologías para movilidad IP de servidores), y las técnicas a utilizar, sólo queda por definir las medidas con las que se realizaría el trabajo. Se eligieron distintas pruebas para poder comparar las tecnologías desde diversos puntos de vista. Estas pruebas fueron:

Ping: Utilidad de diagnóstico por excelencia. Consiste en el envío de paquetes de datos de un host a otro, con el objetivo de ver el tiempo que tarda un host en recibir la petición, y en devolverla. Para esta prueba, se utilizaron diversos tamaños de paquete, y se aumentó la frecuencia de envío, para analizar el comportamiento de las tecnologías.

Peticiones http: el protocolo por excelencia de internet es el http. “Protocolo de transferencia de hipertexto”, el acceso a la información de las páginas web suele desarrollarse por éste método. Con esta prueba se podría comprobar el comportamiento de usuarios lanzando peticiones sobre el servidor para la consulta de hipertexto, y si se recuperaría de la marcha de un servidor de una red a otra mientras se realizan las consultas.

Descarga de un fichero por http: Otro de los usos básicos de internet es la descarga de ficheros a través de las páginas web. Un servidor con ficheros permite la descarga de ficheros a través del protocolo http de diversos clientes. Se quiere estudiar el comportamiento de las conexiones durante el cambio de red de un servidor.

Transferencia de un fichero por scp: Scp es un medio de transferencia segura que usa el protocolo ssh. Esta transferencia se realiza entre dos host en remoto, o entre un host local y otro host remoto. Durante la transferencia de datos, los datos son cifrados para evitar posibles extracciones de la información por agentes externos a la comunicación. Uno de los objetivos de las mediciones era ver que ocurría cuando, en mitad de una descarga, el servidor cambiaba su IP, para observar si se cancelaba la conexión, se paraba un tiempo y luego proseguía, si había que reactivar la conexión, etc...

Transferencia de un fichero por ftp: Ftp es el protocolo de transferencia de ficheros por excelencia. Utiliza para ello habitualmente el puerto 20. Se basa en la transferencia de ficheros entre un servidor y un cliente. El cliente puede coger un fichero desde el servidor, o puede depositar un fichero en el mismo. Para ello, deberá loggearse antes contra el servidor, y una vez que ya esté realizada la conexión, podrá empezar la descarga del fichero. FTP está orientado a conseguir la mayor velocidad de transferencia, pero no es tan seguro como lo es scp. Al igual que antes, se realizará un cambio de la red el servidor durante la descarga, para estudiar su comportamiento.

Escaneo de puertos con hping: Hping es una herramienta para realizar auditorías y poder realizar pruebas sobre una red determinada. Además del envío de paquetes, tanto a nivel tcp, como a nivel udp, tiene una aplicación interesante que todavía no se ha mencionado, y es el escaneo de puertos. Con hping, se puede enviar paquetes a cada puerto, para comprobar qué puertos están abiertos y cuáles no, dependiendo de la respuesta obtenida.

Comparativa general en entorno virtual

		HIP	VPN
PING	Altera la pila de protocolo	Si	No
	Tiempo de reconfiguración	12.1s	5.2s
	Tiempo de más necesitado	1.13	4.65
PETICIONES WEB	Tiempo de reconfiguración	16.16s	7.5
	Tiempo de más necesitado	1.05	1.73
DESCARGA HTTP	Tiempo de reconfiguración	16.16s	7.47s
	Tiempo de más necesitado	2.11	54.1
SCP	Tiempo de reconfiguración	19.56s	13s
	Tiempo de más necesitado	-1.29*	66.9
FTP	Tiempo de reconfiguración	No medido	No medido
	Tiempo de más necesitado	9.75	9.5
HPING	Tiempo de reconfiguración	No medido	No medido
	Tiempo de más necesitado	1.31	3.33

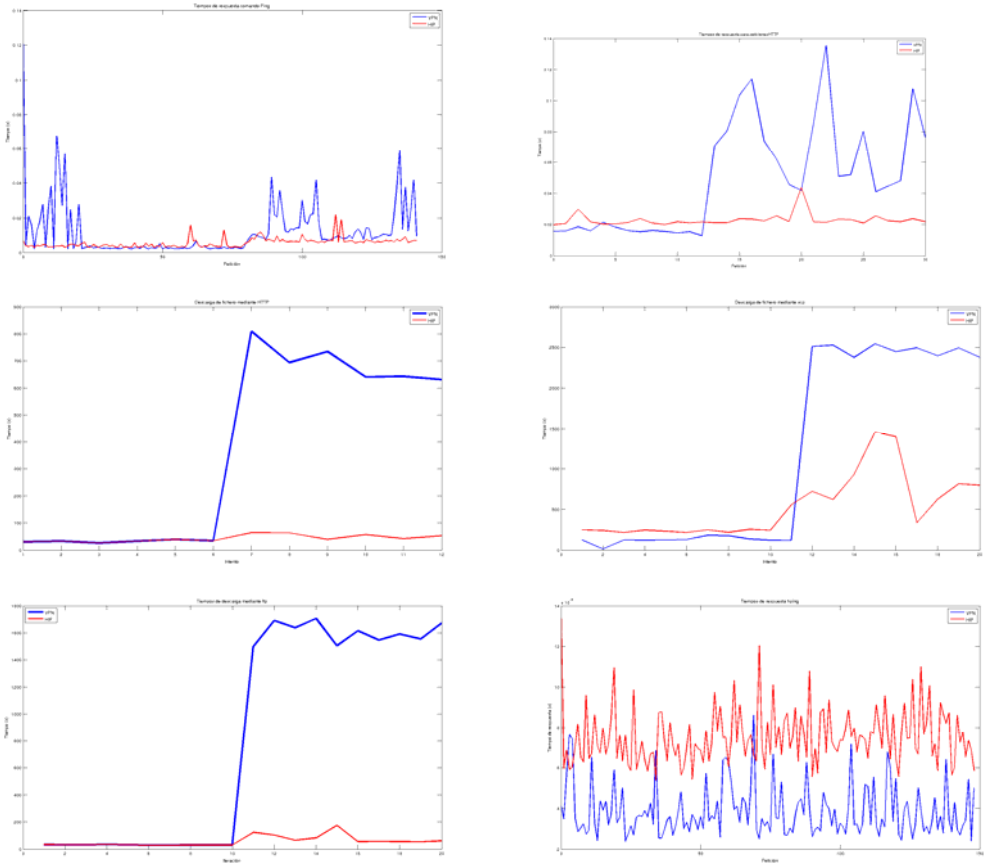
Comparativa general en entorno real

		HIP	VPN
	Altera la pila de protocolo	Sí	No
PING	Tiempo de reconfiguración	22.5s	29.1s
	Tiempo de más necesitado	1.77	2.37
PETICIONES WEB	Tiempo de reconfiguración	20s	40s
	Tiempo de más necesitado	1.06	2.75
DESCARGA HTTP	Tiempo de reconfiguración	31.69s	19.42s
	Tiempo de más necesitado	1.93	21.29
SCP	Tiempo de reconfiguración	34.53s	1m 49s
	Tiempo de más necesitado	3.62	18.67
FTP	Tiempo de reconfiguración	No medido	No medido
	Tiempo de más necesitado	2.63	52.49

Gráficos de tiempos de las medias

Las pruebas realizadas durante el transcurso del proyecto han servido de base para hacer las comparaciones que se necesitaban para dar respuesta a la pregunta que quería responderse en este proyecto. Dado que el objetivo era evaluar las dos tecnologías mencionadas y decir cuál se comporta mejor en la mayoría de los casos, la comparación de resultados obtenidos ha sido fundamental.

Las siguientes imágenes muestran de forma gráfica las comparaciones realizadas, que son gráficas que especifican tiempos de respuesta y de transferencia de ficheros. Las líneas azules corresponden a OpenVPN y las rojas a HIP. Las gráficas se corresponden, respectivamente, a las pruebas sobre ping, peticiones http, descarga mediante http, scp, ftp y escaneo de puertos mediante hping.



En la mayoría de los casos las líneas azules van por encima de las rojas, lo que permite ver que los tiempos obtenidos con OpenVPN son peores. Este hecho se desarrolla más a fondo en las conclusiones.

Conclusiones

Conclusiones técnicas

Vemos que si quisiéramos usar estas tecnologías para no perder conectividad en ningún momento probablemente usaríamos HIP: es más rápido una vez cambiado de red y no tiene mucho que envidiar en la velocidad de reconfiguración si comparamos con VPN.

¿Pero cuáles son las ventajas y los contras de estas dos tecnologías?

VPN provee seguridad, estabilidad y comprobados mecanismos de cifrado sin sufrir la complejidad de otras soluciones VPN como las de IPsec. Ofrece también la posibilidad de de implementar dos modos básicos en capa 2 o capa 3 con lo que somos capaces de enviar información en otros protocolos no-IP y también ofrecer soporte proxy.

OpenVPN soporta también una amplia gama de modos de configuraciones, incluyendo acceso remoto, sitio a sitio VPN, Wi-Fi, balanceo de carga y conmutación por error.

HIP en cambio propone una nueva capa en la pila de protocolos, entre los niveles de red y de transporte. Este nuevo nivel proporciona autenticación, con independencia de la dirección IP del host. Además, con el protocolo HIP se establece una asociación segura entre los dos hosts que participan en la comunicación. Esta asociación, que se crea como paso previo al intercambio de datos, proporciona autenticación, integridad, confidencialidad de los datos enviados y es resistente a ataques de Denegación de Servicio (DoS) y a ataques de reenvío de paquetes.

HIP se encarga de que los usuarios no pierdan la identidad que poseen aunque cambien de ubicación en la red o cambien de dirección IP. Estos cambios son totalmente transparentes para el usuario si utiliza HIP.

HIP también añade características multi-homing de movilidad a los dispositivos habilitados por HIP, es decir, que se tiene más de una conexión simultánea a Internet. En HIP un nodo es capaz de recibir paquetes creados por HIP de cualquier dirección IP. Un nodo puede cambiar su ubicación topológica y continuamente enviar y recibir paquetes desde y hacia sus compañeros

Estas tecnologías nos permiten seguir conectados a una red en la que físicamente no lo estamos y es una forma de abordar el problema de la movilidad en internet. A la hora de comparar estas dos tecnologías y dar un veredicto, saltan las dudas puesto que cada tecnología tiene sus pros y sus contras. Open VPN tiene unas velocidades de reconexión muy buenas comparadas con hip: Open vpn ronda los 9-16 segundos en máquinas virtuales y 20-30 en la infraestructura real cuando hip dobla estas velocidades. Si es cierto que la velocidad de Open VPN en transferencia de archivos grandes deja mucho que desear: por ejemplo un archivo de 200 megas, que equivaldría a un video en internet, se descargaría a 100 KB/s, una

velocidad muy lenta hoy en día si lo comparamos con la banda ancha. Hip en cambio nos ofrece velocidades de entre 300 y 700 KB/s, algo más aceptables. La perdida de paquetes también es inferior si usamos el protocolo HIP. Se pierden entre 1 y 3 paquetes en la mayoría de pruebas mientras que Open vpn puede llegar a perder 7 paquetes en un entorno real. Como hoy en día todo se rige por la velocidad y el tiempo vale oro en internet, a las dos tecnologías les hace falta un empujón, por eso mismo dejamos que ustedes prueben que tecnología les gustan más.

Aún así, en principio nos decantamos más por HIP, ya que como se ha dicho anteriormente consigue mayores velocidades a pesar del hándicap a la hora de realizar los cambios de red. Nos parece que no tiene sentido utilizar VPN y esgrimir como ventaja el hecho de que los cambios de red se den más rápidamente, si después no se va a poder una conexión aceptable que posibilite casi cualquier tarea en la red de hoy en día.

Como conclusión, por tanto, nos gustaría destacar que ambas tienen sus ventajas e inconvenientes, pero que si hay que decantarse por una nos quedamos con HIP.

Conclusiones personales

En primer lugar, hay que decir que la temática del proyecto ha sido en cierto modo diferente a lo que se ha hecho en años anteriores, lo cual ha servido de herramienta muy valiosa para el aprendizaje en un tema como son las redes. Los componentes del grupo pensamos que como ingenieros que queremos ser, el campo de las redes es un tema que debemos dominar y es precisamente esto lo que hemos conseguido con el proyecto. Si bien ya habíamos cursado la asignatura de redes, este proyecto nos ha puesto ante un reto que escapaba de los conceptos teóricos cursados en dicha asignatura, para adentrarnos en un entorno más práctico. Podemos decir que hemos aprendido a configurar y poner en marcha routers, establecer infraestructuras VPN y HIP e incluso hemos obtenido conceptos teóricos de Mobile IP, a pesar de no haber podido desarrollar este protocolo como nos hubiera gustado.

En lo personal, nos hemos visto en la necesidad de trabajar en grupo, organizando las tareas a realizar para cada integrante del equipo y trabajando para que el trabajo en paralelo se sincronizara de la mejor manera posible. Queremos decir que esta necesidad nos ha dado más experiencia aún en trabajar en grupo con todas las ventajas e inconvenientes que ello conlleva, pero también queremos hacer un poco de autocrítica, ya que pensamos que en algunas fases del proyecto la organización del grupo podría haber sido mejor. Pese a esto, estamos satisfechos del trabajo realizado y en general creemos que hemos sido capaces de hacer frente a las dificultades que se nos han planteado.

Además del trabajo en grupo, la propia duración del proyecto ha sido un escollo más en este tiempo, ya que al no ser un período parecido al de prácticas que hayamos podido realizar en anteriores cursos, nos ha sido más complicado dimensionar en el tiempo todas las tareas que teníamos para hacer.

Para terminar con estas conclusiones personales, por tanto, nos gustaría hacer hincapié en que nos ha parecido un proyecto que se ha alejado de un proyecto más “típico”, como podría ser el desarrollo de un software. Sin intentar menospreciar a este tipo de proyectos,

queremos resaltar que la temática en la que se ha desarrollado el nuestro nos ha permitido aprender sobre un tema que nos parece muy importante, y que a lo largo de la carrera no hemos tratado tan a fondo como otras materias.

ANEXOS

Anexo I: Manual de instalación de OpenVPN

Para realizar las pruebas de vpn, instalamos en las maquinas virtuales **OpenVPN**

OpenVPN es una aplicación de software de código libre y abierto que implementa red privada virtual (VPN) para crear soluciones de seguridad punto a punto o conexiones de sitio a sitio (http://www.openvpn.net/), una aplicación de software gratuito, de código libre y abierto, que implementa vpn. Se implementa en las capas 2 y 3 del modelo OSI, usa protocolos SSL/TLS, y se basa en la certificación del cliente en el servidor mediante una serie de certificados. No es un proxy de aplicación web, ni trabaja a través de un navegador web, sino que crea soluciones de seguridad punto a punto, o conexiones a sitio. Para ello, crea una interfaz en el cliente, y le asigna una IP de la red privada del servidor, con lo que ahora el cliente puede trabajar con los recursos locales, tales como impresoras, servidores…, sin necesidad de estar en la red físicamente.

INSTALACIÓN DE OPENVPN

Para iniciar la instalación de openvpn trabajamos con dos máquinas, una que hiciera de servidor OpenVpn, y otra que hiciera de cliente; la primera sería la encargada de otorgar una IP de su red Privada al cliente, que estaba fuera de esta.

Para llevar a cabo la instalación, buscamos en diversas páginas de internet (ver bibliografía), y contamos con una configuración de nuestro profesor, Rafael Moreno. A partir de este material, realizamos la instalación siguiendo los siguientes pasos:

1.- Instalamos openvpn: en un terminal, escribimos:

sudo apt-get install openvpn

2.- Copiamos toda la información necesaria a un directorio sobre el que desarrollaremos todo el trabajo, el /etc/openvpn/ ; esto se hace para que futuras actualizaciones de openvpn no sobrescriban nuestras modificaciones.

cd /etc/openvpn/

cp -r /usr/share/doc/openvpn/examples/easy-rsa/2.0/ /etc/openvpn/*

3.- Modificar el fichero vars (/etc/openvpn/vars) para añadir nuestros datos a los cinco campos necesarios: KEY_COUNTRY, KEY_PROVINCE, KEY_CITY, KEY_ORG, and KEY_EMAIL. Para ello, abrimos el fichero con cualquier editor (gedit vars), y ponemos nuestro país, nuestra provincia, nuestra ciudad, nuestra organización y nuestro email. Es importante rellenar estos datos, no dejar nada en blanco.

4.- Inicializamos el certificado de autorización y la infraestructura de la clave de acceso pública. Para ello, realizamos la siguiente secuencia de órdenes:

. ./vars
./clean-all
./build-ca

Esta última instrucción generará el certificado de autorización CA. Para ello nos pedirá unos serie de datos, de los cuales, unos ya están definidos en vars, y otros habrá que introducirlos ahora, como son nombre y nombre de la sección.

5.- Ahora es el momento de crear los certificados, tanto del servidor como de los clientes. Todo se va a crear en la máquina del servidor, para asegurarnos que las claves tengan la misma certificación. Primero, creamos los del servidor. Ejecutamos en consola:

./build-key-server server

Server es el nombre que damos a los permisos, tanto al crt como a la key del servidor. Podría haber sido cualquier nombre. Después de ejecutarlo, nos preguntará el "common name", y habrá que ponerle el mismo, server, que hayamos puesto a la ejecución del comando. Por último, nos pedirá confirmación a dos preguntas ("Sign the certificate?" y "1 out of 1 certificate requests certified, commit?"). A ambas preguntas responderemos afirmativamente (y), con lo que tendremos generados los ficheros necesarios para el servidor.

Pasamos ahora a los certificados de los clientes. Para ello, ejecutamos el siguiente comando:

./build-key cliente1

Este nombre (cliente1) deberá ser el mismo que luego introduzcamos en common name, y debe ser único para cada cliente que quiera loggearse en el servidor.

6.- Generar los parámetros "Diffie Hellman", o dh. Este fichero indica el método de intercambio de claves y autentificación utilizado por el servidor. Para ello, ejecutamos el siguiente comando:

./build-dh

Ya tenemos instalado openvpn en ambas máquinas. Ahora es el momento de mover los certificados a sus sitios correspondientes.

- ca.crt debe estar tanto en el servidor como en los clientes
- ca.key, dh{n}.pem, server.crt y server.key solo en el servidor
- client1.crt y client1.key solo en el cliente correspondiente.

Antes de arrancar el servicio, tenemos que modificar los ficheros de configuración de ambas máquinas. Aquí se puede ver la configuración que usamos para la primera prueba, los cuales tenemos de muestra en /usr/share/doc/openvpn/examples/sample-config-files:

Descomprimir el server.conf.gz

sudo cp /usr/share/doc/openvpn/examples/sample-config-files/server.conf.gz /etc/openvpn/
sudo gzip -d /etc/openvpn/server.conf.gz

server.conf

```
port 1194 // puerto por defecto para openvpn
proto udp // el protocolo es udp, no tcp.
dev tap0 // interfaz virtual sobre la que se va a comunicar el cliente.
ca /etc/openvpn/ca.crt // fichero de autentificación.
cert /etc/openvpn/server.crt
key /etc/openvpn/server.key    // certificado y clave del servidor.
dh dh1024.pem // fichero con los parámetros Diffie Hellman
server-bridge 192.168.57.101 255.255.255.0 192.168.57.150 192.168.57.200
        // Indica la ip PRIVADA del servidor, la máscara de red, y el rango de ip´s que
           otorga a los clientes de openvpn. Esta ip irá en el tap del cliente.
duplicate-cn
keepalive 10 120
cipher AES-128-CBC // tipo de codificación de las contraseñas.
comp-lzo
persist-key
persist-tun
status openvpn-status.log
log       openvpn.log // para la salida de los mensajes
verb 9 // nivel de los mensajes que lanza openvpn
mute 20
```

cliente.conf

```
client

dev tap0 // la misma que se creará en el servidor
proto udp // el protocolo es udp, no tcp.
remote 192.168.56.101 1194 // ip PUBLICA del servidor, y el nº del puerto
user root
group root
persist-key
persist-tun
ca /etc/openvpn/ca.crt // ficherode autentificación.
cert /etc/openvpn/cliente1.crt
key /etc/openvpn/cliente1.key    // certificado y clave del servidor.
cipher AES-128-CBC // tipo de codificación de las contraseñas.
comp-lzo
verb 3
mute 20
```

Por último, tenemos que crear y levantar el tap que irá sobre el bridge en el servidor openVPN (solo en el servidor, no en los clientes). Para iniciar openvpn, primero tenemos que iniciar el tap. La secuencia correcta es la siguiente:

- run **bridge-start**
- run openvpn
- stop openvpn
- run **bridge-stop**

Ahora tenemos que definir un script para bridge-start y bridge-stop. Para ello, podemos ayudarnos de los script de ejemplo que tenemos en */usr/share/doc/openvpn/examples/sample-scripts*. Ahí tenemos dos ficheros, bridge-start y bridge-stop, los cuales editaremos para que el bridge y el tap estén sobre la interfaz de la parte privada del servidor. Aquí tenemos un ejemplo de la configuración.

bridge-start

```
#!/bin/bash

###################################
# Set up Ethernet bridge on Linux
# Requires: bridge-utils
###################################

# Define Bridge Interface
br="br0"  //bridge que queremos crear

# Define list of TAP interfaces to be bridged,
# for example tap="tap0 tap1 tap2".
tap="tap0" // tap que queremos crear, y que irá sobre el bridge

# Define physical Ethernet interface to be bridged
# with TAP interface(s) above.
eth="eth2" // interfaz de la red privada del servidor openvpn
eth_ip="192.168.57.101" // ip privada del servidor openvpn
eth_netmask="255.255.255.0"
eth_broadcast="192.168.57.255"

for t in $tap; do
        openvpn --mktun --dev $t
done

brctl addbr $br
brctl addif $br $eth
```

```
for t in $tap; do
        brctl addif $br $t
done

for t in $tap; do
        ifconfig $t 0.0.0.0 promisc up
done

ifconfig $eth 0.0.0.0 promisc up
```

bridge-stop

```
#!/bin/bash

####################################
# Tear Down Ethernet bridge on Linux
####################################

# Define Bridge Interface
br="br0" // bridge creado anteriormente

# Define list of TAP interfaces to be bridged together
tap="tap0" // tap creado anteriormente

ifconfig $br down
brctl delbr $br

for t in $tap; do
        openvpn --rmtun --dev $t
done
```

Una vez que ya tenemos la configuración de ambos ficheros, ejecutamos

./bridge-start

Con esto, si hacemos un ifconfig ya veremos que está levantado el tap sobre el bridge, y que el bridge tiene la ip de la interfaz privada.

Ya tenemos todo listo para arrancar el servicio openvpn en ambos sitios. Para ello, lo arrancamos primero en el servidor:

openvpn server.conf

Ahora el servicio está corriendo, y solo falta arrancarlo en el cliente. En una consola de ejecución del cliente, ejecutamos:

openvpn cliente.conf

Ahora, cuando hagamos un ifconfig sobre el cliente, veremos que ha obtenido una dirección ip otorgada por el servidor de su parte privada, por lo que ya tenemos el objetivo que queríamos conseguir, tener funcionando openvpn en nuestras máquinas virtuales.

Anexo II: Manual de instalación de Mobile IP

Este documento detalla cómo se ha hecho la instalación de la implementación del protocolo Mobile IP llamada Dybamics. En primer lugar se hace una introducción a Dynamics y en los siguientes apartados se detallará, en primer lugar, los requisitos del sistema para que Dynamics pueda funcionar y seguidamente se explicarán los pasos a seguir para instalar la mencionada implementación de Mobile IP.

Introducción a Dynamics

El sistema de Mobile IP Dynamics, originalmente desarrollado en la universidad de Helsinki (Helsinki University of Technology), es un software Mobile IP para Linux que se caracteriza por ser escalable, dinámico y jerárquico. Además, el Nodo Movil (Mobile Node) de Dybamics ha sido portado parcialmente al sistema operativo Microsoft Windows (98SE, ME, NT4, 2000).

Requisitos

Para poder utilizar Dynamics en una máquina, esta debe cumplir ciertos requisitos que se detallarán en el presente apartado.

Dynamics ha sido desarrollado para ser utilizado bajo el sistema operativo Linux, concretamente para versiones del kernel superiores a 2.2.x, aunque es posible que también funcione en algunos kernel de la versión 2.1.x. Además de la versión correcta, los distintos componentes que utiliza Dynamics necesitan que los kernels tengan activadas algunas opciones del kernel. En primer lugar, la máquina que ejecutará el Agente Externo o Foreign Agent necesita advanced policy routing (advanced routing: CONFIG_IP_ADVANCED_ROUTER y policy routing: CONFIG_IP_MULTIPLE_TABLES). Por otra parte, el Nodo Móvil o Mobile Node funcionará de forma más eficiente con Linux Socket Filters (CONFIG_FILTER).

Para tener un kernel compatible, se requieren las siguientes opciones en tiempo de compilación:

- Soporte para modelos cargables (si el tunneling ipip es utilizado como módulo)
- Opciones de red (in addition to the default selections):
    - Packet socket (CONFIG_PACKET)
    - Kernel/User netlink socket (CONFIG_NETLINK)
    - Routing messages (CONFIG_RTNETLINK)
    - IP: Socket Filtering (CONFIG_FILTER)

    - IP: tunneling (CONFIG_NET_IPIP)
- Además, se requieren las siguientes opciones para los Foreign Agents:
    - IP: advanced router (CONFIG_IP_ADVANCED_ROUTER)
    - IP: policy routing (CONFIG_IP_MULTIPLE_TABLES)
- Opciones adicionales para la extensión inalámbrica para Mobile Node:
    - Wireless LAN (non-hamradio) (CONFIG_NET_RADIO)

Para poder proceder a la compilación de los programas que vienen con Dynamics, es necesario tener instalada la librería GNU MP.

Para terminar con el apartado de requisitos, se pasarán a detallar los requisitos de hardware que exige el sistema y una lista de procesadores en los que Dynamics ha sido probado con éxito.

Requisitos hardware:

Espacio en disco:
Distribución de código fuente: 1.5 MB
Binarios del sistema y testeo, manuales y ficheros de configuración: 1 MB
Espacio en disco necesario para la compilación: menos que 5 MB
Sistema con full debugging:9 MB
Utilización de la memoria:
Menos de 1 MB por demonio.

Lista de procesadores en los que se ha probado Dynamics:
Arquitectura Intel ix8
Digital Alpha (64-bit)
Motorola 68k (big endian)
Power PC 603e (big endian)

Instalación

Para proceder a la instalación de Dynamics es preciso descargar el archivo comprimido desde la página del proyecto (http://sourceforge.net/projects/dynamics/) y descomprimirlo en el directorio que se desee.

Como se ha dicho en el apartado de requisitos, para poder proceder con la instalación de Dynamics, hay que instalar previamente las librerías GMP. Para ello se teclea en la terminal el siguiente comando:

```
sudo apt-get install libgmp3-dev
```

Una vez hecho esto, puede instalarse Dynamics. Se accede al directorio en el que se ha descomprimido el fichero descargado y se teclean de uno en uno los siguientes comandos:

```
./configure          (crea algunos ficheros y un makefile)
make                 (utiliza el makefile creado para compilar el paquete)
make check           (hace algunas comprobaciones, creo)
sudo make install    (instala el programa)
```

Una vez hecho esto se supone que todo está instalado correctamente.

A la hora de ejecutar uno de los daemons, hay que cargar primero el modulo IP-within-IP. Para ello debe teclearse lo siguiente:

```
sudo modprobe ipip
```

Una vez cargado dicho modulo, puede iniciarse uno de los daemons, por ejemplo para dynhad (Home Agent Daemon):

```
sudo dynhad --debug --fg --conf /home/markel/Dynamics/src/ha/dynhad.conf
```

El comando --conf es para especificar un fichero de configuración para el daemon. En este caso se le ha dado la ruta al fichero de configuración de ejemplo que viene con Dynamics.

Si todo ha ido correctamente debería de mostrarse un mensaje diciendo que se está utilizando dicho fichero, y teóricamente el Home Agent ya se está ejecutando.

Para el resto de daemons, el proceso debería ser igual cambiando el nombre del daemon en el último comando.

Anexo III: Manual de instalación de HIP

1- INSTALACIÓN (linux)

Nos descargamos el archivo desde: http://sourceforge.net/projects/openhip/files/hip/hip-0.8/

El paquete debian instala lo siguiente:

```
> sudo dpkg -L openhip
/.
/usr
/usr/local
/usr/local/sbin
/usr/local/sbin/hip
/usr/local/sbin/hitgen
/usr/local/etc
/usr/local/etc/hip
/usr/local/etc/hip/known_host_identities.xml
/usr/share
/usr/share/doc
/usr/share/doc/openhip-0.5
/usr/share/doc/openhip-0.5/AUTHORS
/usr/share/doc/openhip-0.5/README
/usr/share/doc/openhip-0.5/COPYING
```

Desinstalar

Podremos desinstalar Hip en un futuro con el siguiente comando:

```
sudo apt-get remove openhip
```

Configuración

Una vez instalado el paquete, procedemos a su configuración. Para ello usamos los comandos:

cd /usr/local/sbin/

```
./hitgen
./hitgen -conf
./hitgen -publish
```

Haciendo esto generaremos los siguientes archivos:

/usr/local/etc/hip/my_host_identities.xml /usr/local/etc/hip/hip.conf /usr/local/etc/hip/<hostname>_host_identities.pub.xml

Con ./hitgen creamos my_host_identities.xml que contiene el host identity y el HIT de la máquina.
./hitgen vale para generar el hip.conf y con -publish generamos el archivo que tendremos que copiar en el archivo known_host_identities de la máquina a conectarnos.

Ejemplo de <hostname>_host_identities.pub.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<!--The following HITs can be copied into a known_host_identities.xml file.-->
<known_host_identities>
  <host_identity alg="RSA" alg_id="5" length="128" anon="no" incoming="yes">
    <name>ibai-Inspiron-1750-1024</name>
    <HIT>2001:1c:e434:1461:9c57:61e8:dd9f:21c6</HIT>
    <LSI>1.159.33.198</LSI>
  </host_identity>
```

El contenido de este archivo tendremos que copiar en el known_host_identities.xml de la máquina remota. Este archivo tiene el siguiente formato:

```
<name> nombre de la máquina</name>
<addr> IP de la máquina</addr>
<HIT> HIT de la máquina</HIT>
<LSI> LSI de la máquina</LSI>
```

Nosotros hemos copiado el de la máquina remota en nuestro known_host identities quedando de esta forma:

```
<?xml version="1.0" encoding="UTF-8"?>
<!--This file has been saved by the HIP daemon. User edits may be lost
    when the HIP daemon terminates. Comments are not preserved. -->
<known_host_identities>
  <host_identity alg="RSA" alg_id="5" length="128" anon="no" incoming="yes">
    <name>Pc-Javi-1024</name>
    <addr>192.168.56.103</addr>
    <HIT>2001:1e:b40:b3cc:86c7:7dd1:4b69:3db8</HIT>
    <LSI>1.105.61.184</LSI>
```

```
  </host_identity>
</known_host_identities>
```

La configuración de nuestras máquinas ha sido de la siguiente forma:

3 máquinas: servidor, cliente y router. El router está conectado a las dos redes y bien el servidor como el cliente están conectados en un principio a la red 192,168,56,0. En las pruebas, el cliente se traslada a la red 192,168,57,0.

Ejecución

/usr/local/sbin/hip -v (en modo verbose)